ESD-TR-71-370, Vol. I

SECURITY OF THE TACC DATA BASE STUDY (DESCRIPTION OF AUTOMATIC DATA BASE SECURITY TECHNIQUES)

September 1970

DEPUTY FOR PLANNING AND TECHNOLOGY
HQ ELECTRONIC SYSTEMS DIVISION (AFSC)
L. G. Hanscom Field, Bedford, Massachusetts 01730

(Prepared under Contract No. F19628-70-C-0185 by Hughes Aircraft Company, Ground Systems Group, Fullerton, California 92634.)

LEGAL NOTICE

OTHER NOTICES

ESD-TR-71-370, Vol. I

SECURITY OF THE TACC DATA BASE STUDY (DESCRIPTION OF AUTOMATIC DATA BASE SECURITY TECHNIQUES)

September 1970

DEPUTY FOR PLANNING AND TECHNOLOGY
HQ ELECTRONIC SYSTEMS DIVISION (AFSC)
L. G. Hanscom Field, Bedford, Massachusetts 01730

(Prepared under Contract No. F19628-70-C-0185 by Hughes Aircraft Company, Ground Systems Group, Fullerton, California 92634.)

LEGAL NOTICE

OTHER NOTICES

ABSTRACT

This report presents the results of a study which surveyed the various aspects of system security hardware, software and procedural techniques in use in current and proposed automated systems.

Thirty-four systems were considered, including 20 government and 14 commercial systems. The security requirements, system environment and function, and techniques employed are described for each system. A total of 95 techniques are identified, of which 41 are software, 34 are hardware, and 20 are procedural techniques. The total number of techniques found as a result of this survey is a considerably larger number than known to exist prior to initiation of the survey. Some quantitative data detailing the relative costs involved in developing, using, and maintaining these techniques was obtained and is provided in this report. Qualitative estimates of cost are made in the remaining cases.

An initial approach to categorizing systems by their security requirements and categorizing techniques by their application to security requirements was devised and applied to the raw data obtained from the survey. The intention is to present the designers of the Post-1975 TACC with a consolidated source of data concerning security techniques and with a tool to evaluate the data and select the techniques applicable to the Post-75 TACC security requirements.

Key words: Access Control

Privacy

Protection

Security

Security Requirements

CONTENTS

Paragraph	Page
Section 1 — OBJECTIVES AND METHODOLOGY	
A — General	1-1
B — Task I Objective	1-1
C — Task I Methodology	1-2
Section 2 — DETERMINING SYSTEMS SECURITY REQUIREMENTS	
A — Introduction	2-1
B — Statement of the Problem	2-1
C — Security Requirements Common to all Systems	2-3
1 — Support Access	2-3
2 — Failure Access	2-4
3 — Deliberate Passive Access	2-4
D — Security Requirements Particular to a Given System	2-6
1 — User Language Capability	2-6
2 — Terminal Location and Usage	2-8
3 — Data Classification	2-8
E — Summary	2-8
Section 3 — CATEGORIZING GOVERNMENT SYSTEMS BY THEIR SECURITY REQUIREMENTS	
A — Introduction	3-1
B — Common Aspects of Government System	3-1
C — Security Requirements and Major Techniques of the Systems Considered	3-1
1 — Local Terminal Systems	3-3
2 — Remote Terminal Systems	3-5
D — Summary	3-11
Section 4 — CATEGORIZING COMMERCIAL SYSTEMS BY THEIR SECURITY REQUIREMENTS	
A — Introduction	4-1
B — Common Aspects of Commercial Systems	4-1
C — Security Requirements and Major Techniques of the Systems Considered	4-4

1 – Local Terminal Systems 4-4
2 – Remote Terminal Systems 4-5

Section 5 – CATEGORIZING AND DESCRIBING SECURITY TECHNIQUES

A – Introduction 5-1
B – Categorizing Techniques by System Parts 5-1
C – Description of Security Techniques 5-1
1 – Software Techniques 5-1
2 – Hardware Techniques 5-14
3 – Procedural Techniques 5-22
D – Categorizing Techniques by Security Threat 5-23
1 – File and Program Access 5-23
2 – Hardware Access 5-25
3 – Communication Lines Access 5-28
E – Categorizing Techniques by Security Requirements 5-28

Section 6 – DETERMINING TECHNIQUES APPLICABLE TO THE POST-1975 TACC

A – Introduction 6-1
B – TACC Security Requirements 6-1
C – Categories of Techniques Applicable to the TACC 6-1
D – Tradeoffs Among Applicable Techniques 6-4
E – Comparison Between Software, Hardware, and Procedural Techniques 6-12
1 – Hardware/Software Comparisons 6-12
2 – Manual-Automated Procedural Comparisons 6-13

Appendix A – BIBLIOGRAPHY OF APPLICABLE DOCUMENTS

Appendix B – SYSTEMS SURVEY

Appendix C – POST-75 COMPUTER HARDWARE PREDICTED MEAN TIME BETWEEN FAILURES (MTBF)

GLOSSARY OF TERMS USED

SECTION 1

OBJECTIVES AND METHODOLOGY

Section 1

OBJECTIVES AND METHDOLOGY

A. GENERAL

This report presents the results of the Task 1 (Security Techniques Review) study effort as specified in the TACC Data Base Study (Contract Number F19628-70-C-0185). The objective of the contract is to establish a TACC data base baseline from which a clear understanding of security and access requirements can be obtained. This baseline is comprised of two parts:

1. A review of hardware, software, and procedural techniques used to maintain data base security and access limitation control in current and completed automated systems.

2. A tabulation and analysis of the classified data, authorized users of that data and characteristics of both the data and the users as exists in the current tactical data base relevant to the proposed TACC complex.

B. TASK I OBJECTIVE

The objective of Task I is to review current and completed studies which have addressed the security and access limitation problems for automated information systems; analyze the data collected considering the differences in techniques as required by systems, all users, and particular users; and discuss technique for data base security and access control applicable to the Tactical Air Control Center complex.

Guidance as to the direction and expected results of Task 1 is provided in the statement of work and called for accomplishment of the following subtasks:

1. Review current and completed studies addressing security and access limitation in automated systems

2. Limit review to hardware, software and procedures used to maintain data base security

3. Consider remote terminals, communications networks, and computer processing and storage elements

4. Consider both intentional and unintentional alteration or access

5. Describe the particular security requirements and summarize the implementation approach for each study and system reviewed

6. Discuss techniques for data base security and access control that could be applicable to the Tactical Air Control complex

C. TASK I METHODOLOGY

Since Task I has as its primary objective the examination of security-oriented techniques and the application of these techniques to TACC security requirements, it was divided into two principle study phases, a data collection phase and a data analysis phase. In practice, there has been a considerable overlap between the two. A reporting phase consisting of two briefings and this document represent the conclusion of the Task 1 part of this three part study.

1. Data Collection Methodology

Nearly 200 documents selected from Defense Documentation Center (DDC) developed bibliographies, from indicies in the open litreature, and from the private collections of concerned individuals were obtained and reviewed. Many of the documents discuss various effects that (might) result from not having a suitable security (or "protection," or "privacy") scheme, but do not develop systematic requirements or definable solutions to the problems raised. A bibliography of the 65 documents considered most relevant to the study are listed in Appendix A of this report.

For many of the systems reviewed, it was sufficient to examine the available literature, particularly in the case of the standard commercial systems and several of the second and early third generation government systems. Where it seemed that a particular system had implemented or attempted to implement somewhat unique approaches to the security problem or where it was felt that concrete information could be obtained concerning the cost and/or effect of their security schemes, direct interviews with responsible individuals were conducted. In some cases, information was exchanged that would not otherwise have been available.

2. Data Analysis Methodology

The general methodology used in the study was to develop a method to categorize systems by their security requirements, consider what security techniques were applicable to each category, and then to determine the category and techniques applicable to the TACC. In developing the various security requirements categories the following was accomplished:

a. A common way to describe security requirements was developed

b. A way to identify which techniques were applicable to a given security requirement was determined.

c. A careful distinction was made between the threat posed by illegal users and the threat posed by untrustworthy support personnel because of the difference in technique approaches that can be used for the two cases.

d. Those design parameters of an automated system that influence security requirements were identified.

e. Qualitative costs were estimated for techniques since they influence the particularity of their application.

The steps followed in the analysis are covered in the following sections as indicated below:

a. Determine common and unique characteristics or design features of systems reviewed that influence security requirements – Section 2

b. Categorize systems reviewed by their security requirements – Sections 3 and 4

c. Categorize security techniques by their applicability to system parts, security threat, and security requirement – Section 5

d. Determine techniques applicable to the Post 75 TACC – Section 6

SECTION 2

DETERMINING SYSTEM SECURITY REQUIREMENTS

Section 2

DETERMINING SYSTEM SECURITY REQUIREMENTS

A. INTRODUCTION

The basic security requirement in any system is to prevent unauthorized access or change of data while allowing authorized use necessary to accomplish the system's mission. Manual systems require the protection of data only. Automated systems introduce the added problem of protecting the processes, either programs or hardware, that are used to store, access and change the data. The individual security requirements that pertain to each category of automated systems are described below.

B. STATEMENT OF THE PROBLEM

A closer examination of security requirements indicates that they are dependent on the specific type of threat posed to the system. There are three general classes of threats: unintentional, deliberate passive, and deliberate active.

Unintentional threats are those that arise from hardware and software failures and user errors which allow unauthorized but inadvertent access to files or programs. Deliberate passive threats are caused by electromagnetic radiation from the computer hardware and communications, wiretapping, and "bugs." Deliberate active threats are from illegal use of the access capabilities designed into the system for legitimate users to accomplish their mission responsibilities. Examples of this type of threat are (1) using legitimate access to ask or obtain unauthorized date (browsing), (2) masquerading as a legitimate user, (3) using access to the system as support personnel (systems programmer, operator, hardware maintenance, management) to obtain data or create trap doors into the system, (4) tapping into remote terminals to receive "piggy-back" entry with an authorized user, (5) between lines entry, and (6) cancellation of user's sign-off signals to continue operation. These threats are nearly the same for all systems, differing primarily in the degree which system design features allow exploitation.

This potential for exploitation is created at each point in the design where the system is made accessible, i.e., where it becomes possible for the designer, the maintainer, or the user to interact with the system. Since the security requirements depend on the threat of exploitation and the threat of exploitation in turn depends on the particular design features of the system that determine its accessibility, the key to specifying the security requirements for a system lies in an examination of the system's accessibility.

In this section, the design features that can be used to exploit threats are distinguished for all systems (see Figure 2-1). Certain characteristics are shared by every system;

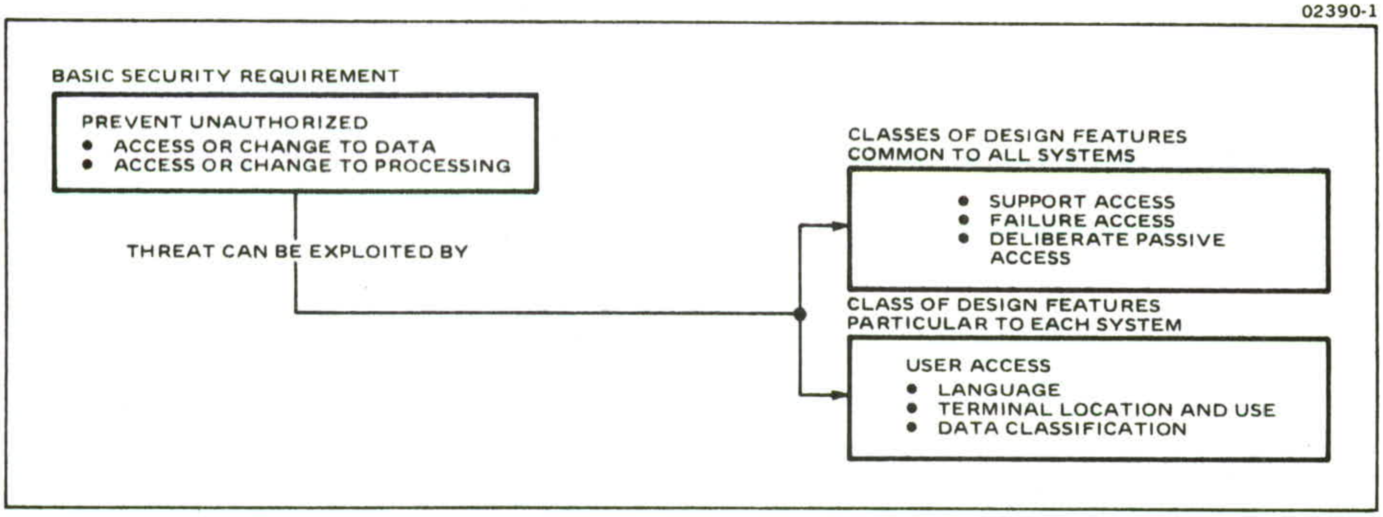

Figure 2-1. Classes of Design Features that Determine System Security Requirements

the need for support access, the reduction of failure and the characteristic of radiation. These are discussed below in the subsections dealing with SECURITY REQUIREMENTS COMMON TO ALL SYSTEMS. In each subsection, the particular security requirements and the techniques that might be used to satisfy these requirements are presented. These requirements are shared by all systems, including the 34 systems reviewed and the TACC. Therefore, the techniques that are applicable in each of these common areas are applicable to every system, including the TACC.

There is another class of characteristics that determine the accessibility provided to the user. They are particular to a given type of system, and can be used to distinguish the accessibility, and consequently the potential threat, of one system from another. In the sections below that deal with SECURITY REQUIREMENTS PARTICULAR TO A GIVEN SYSTEM, three areas of system design are defined that describe the accessibility afforded to the user.

Each area is further subdivided into individual design features and the security requirements associated with each design feature are specified. In Sections 3 and 4, the 34 systems reviewed are assigned to one or another of these categories, depending on the combination of design features that describe the system. Because of the possible variations in the final TACC design, the TACC is tentatively assigned to several categories. The security requirements associated with the design features of a particular category of systems apply to any system assigned to that category.

C. SECURITY REQUIREMENTS COMMON TO ALL SYSTEMS

1. Support Access

All systems have the requirement to allow access for maintenance of the system software and hardware. This "support access" must be provided for the system programmers, maintenance personnel, computer operators, and management personnel responsible for the system operation. It represents a potential means of deliberate active penetration and has been addressed in the literature on non-military systems as the area of most concern. Non-military systems lack the procedural security regulations established by law for military systems.

The support access characteristics were similar in all systems. In order to maintain the system, access at the assembly/POL language level is needed to debug programs, maintain hardware, and establish system operating conditions. Access to the most sensitive programs and tables is necessary to establish access and process rights to programs and data and to service the system. In all government systems, this support activity was conducted only at local terminals within the secure computer area, where common procedural techniques were relied on to limit access to cleared personnel. Effort and responsibility were fragmented in order to isolate and restrict the capability and, hence, security risk associated with any single individual.

Hardware and software security techniques were more an expression of techniques implemented to meet user requirements and the characteristics of the computer equipment installed than a deliberate design to control support access to the system.

The systems reviewed used two levels of protection from compromise: (1) Error hardware interrupts with software maintenance procedures. This was used on second generation machines and provided a limited degree of security protection; (2) third generation machines with privileged mode, privileged instruction set, memory bounds and memory protect hardware. These hardware features, when combined with software partitioned processing and MACRO calls limit the extent of access of support personnel to reasonably defined areas. They also provide an effective method to fragment software maintenance effort and to protect against inadvertant routing errors.

The accessibility afforded to support personnel in commercial systems has received more attention in terms of the development of sophisticated automated security techniques. The primary reason is that military systems have developed strong procedural techniques (clearance procedures for personnel, security regulations) with formal legal penalties for infractions that are not available for use in commercial systems. Consequently, the threat from support personnel is more immediate than in military systems.

Security requirements determined by the need for support access include conventional procedural requirements to identify and certify personnel, as well as 1) the ability to isolate access to programs and data to only those authorized to maintain the particular program or file, 2) the ability to isolate maintenance personnel to the maintenance of specified equipment items assigned and to restrict their use to only specified maintenance software routines, 3) the need for procedures to insure that programs are completely debugged, 4) the need to audit files for unauthorized changes, 5) the ability to determine if equipment is operating properly before it is placed in system used, 6) the ability to detect and control changes to system routines, 7) the ability to bound dumps of memory and peripherals, 8) the ability to determine that a program only performs the function for which it was designed, and 9) the ability to restrict access by internal control tables.

Table 2-2 summarizes the major security techniques implemented or discussed by the systems surveyed in the area of support control. A detailed explanation of the techniques identified is presented in Section 5.

2. Failure Access

The threat of compromise from the release of data or programs due to hardware or software failure is common to all systems and represents a potential means of unintentional penetration.

Failure control of systems is needed to prevent unintentional release of data or processes caused by system hardware or software failure. As in support control, the techniques used by most systems were not implemented as a direct consideration of the security requirements. Rather, in this case, they were designed to meet reliability requirements. Three levels of protection apply to this area: (1) second generation machine techniques for failure control; (2) third generation machine failure control technique; and (3) techniques over and above those implemented that provide increased capability to prevent failure in hardware and software from inadvertantly releasing data.

The hardware mean time between failure (MTBF) predictions for the Post '75 time period give an indication of the degree of the hardware failure problem. MTBF's of from four months to one year are predicted for key circuits that could result in the unintentional release of data (see Appendix C). Parity-check circuits have been implemented in third generation computer systems and the use of additional parity check circuits plus key redundant circuits have been suggested but not implemented to further reduce the possibility of unintentional release of data. Periodic maintenance and hardware on-line and off-line diagnostics are also used by all systems for failure control.

Failure control of software is mainly procedural control or debug discipline and is used by all systems. In third generation computers, the processor mode and memory partitioning help to isolate software failures to a particular job or task. Time limits and line limits are used to recover from software failure by most systems. They provide a technique to limit the effect of software failures. Software on-line and off-line diagnostic programs are also used to limit the effect of software failure. Security requirements determined by the need for failure control include 1) the ability to trap to software error routines when parity errors are encountered, 2) the ability to prevent circumvention by software "bugs" of the partitioning technique that isolates data and programs from unauthorized access, 3) the need to check out and certify program changes and equipment repairs to ensure that they are operating correctly, 4) the need to maintain the protection mechanism when a system error is encountered, 5) the need to recover from failure without revealing protected data and system tables, 6) the need to protect backup files, and to certify that the appropriate backup file is loaded. Table 2-3 summarizes the major failure control techniques and the systems that use the particular technique. A detailed explanation of each technique is contained in Section 5.

3. Deliberate Passive Access

Electromagnetic radiation, wiretapping, and "bugs" can be used on all systems if proper security techniques are not implemented. Electromagnetic radiations from computer equipment power lines and communications lines can be detected and decoded into the binary bits representing the data stream. Wiretapping into communications lines can be used to send and receive data. The planting of "bugs" or recording devices

TABLE 2-2. SECURITY TECHNIQUES APPLICABLE TO THREAT POSED BY SUPPORT ACCESS

Systems	Techniques		
	Hardware	Software	Procedural
• 407L (CRC) • 407L (CRC) • NTDS • 465 L • USN-LCC	• Error Interrupts	• Software Diagnostics	• Security Discipline • Secure Area • Personnel Cleared • Fragment Effort and Responsibility • Security Regulations and Procedures
All other systems surveyed	• Error Interrupts • Privileged Mode • Privileged Instruction Set • Memory Bounds • Read/Write/Execute Bits	• Software Diagnostics • MACRO Calls • Non-Functional Software Design	
No systems	• Dedicated Memory • Ring Segment Bits	• Module Checksum • Module Dialog • Non-Reverse Transforms • Redundant Load • Code Interpreter • Ring Structures	

is possible if proper area security precautions are not used. The only techniques that are applicable to this category of requirements are cryptographic data transformations and/or shielded lines for communication links, and maintenance of a properly secured area. Each of the government systems surveyed (with the exception of COHERENT and NCIC) employ these same techniques. Security requirements determined by the need to prevent deliberate passive access include conventional red/black isolation requirements, as well as 1) the ability to decode transmitted data so that it cannot be deciphered, 2) the need to certify that hardware either cannot, or has not been tampered with, and 3) the ability to make stored data unintelligible to direct dumping.

D. SECURITY REQUIREMENTS PARTICULAR TO A GIVEN SYSTEM

Systems differ in their interface with the system user. The user of the system exercises the system for its functional purpose and is not concerned with its design, implementation, or maintenance. His use of the system is limited to direct accomplishment of the purpose of the system whether it be providing generalized problem solving or accomplishing a specific military mission. User accessibility to a system is defined by the type of system interface, the language capability offered, and the clearance of data and users provided. Different combinations of these imply increasingly sophisticated levels of access rights and hence different possibilities of penetration attempts. User access capability that directly relate to security requirements are discussed in the following paragraphs as (1) language capability, (2) terminal location and usage, and (3) user and data clearance levels. The security requirements of each design feature are indicated in Table 2-4.

1. User Language Capability

The user interfaces with the automated system in either an off-line or an on-line mode. In an off-line mode he submits requests for data services to support personnel and receives as his output printed reports. This mode of operation is typical of closed-shop batch systems. Their security requirements in so far as they concern the user differ from manual systems only by the addition of a requirement for security within the secure area of the computer system.

In the on-line mode, the user is provided a capability to request data services directly from the computer equipment by means of some input device. His form of dialogue (language) with the computer can vary from (1) rigid requests for predetermined fixed transaction input/output, (2) use of a freeform query language, and (3) entry of actual computer programs in procedure-oriented (POL) or assembly language. Each level of language capability implies increased security requirements as the level of capability to access data increases.

Fixed transaction input/output allows the system designer to predetermine what will be the specific input and output allowed for a given user at a given terminal. Using Free Form Query capabilities, the user could possibly ask for data not needed for his mission. It then becomes necessary to increase the security requirements of the system to handle this eventuality and to prevent attempts to deliberate circumvention arising from the increased language capability. The ability to enter POL or assembly language programs places the user at almost the same capability as support personnel and could allow circumvention of security techniques implemented for fixed format or free form query capabilities. A higher level of security requirements is necessary to provide protection against this increased language capability.

2. Terminal Location and Usage

The ability of a user to access or change data from remote terminals suggests penetration methods not possible in a system with only local terminals within a secure area. The communications lines must be protected, system or user errors could allow release of data outside the secure area, and the vulnerability of remote terminal secure areas, especially in a tactical military system, is greater. The use of only local terminals within a secure area allows the centralization of security precautions and the centralized area is usually in a more secure physical area in a tactical environment. There is less risk to the imposter associated with penetration attempts at a remote terminal than within a centralized secure area. Therefore, the addition of remote terminals to a system increases significantly the security requirements of the system.

System security requirements are also influenced by the use of the terminals; that is, whether there is only one class of need-to-know at a given terminal or whether there are multiple classes of need-to-know at a given terminal. Multiple need-to-know at a given terminal requires that the system be able to identify the different user classes at a given terminal and provide protection against "browsing" and "masquerading." The amount of data potentially available at any one terminal is usually greater than that at a single need-to-know terminal.

3. Data Classification

The security classification of data is an expression of the value of the information to national defense and hence the seriousness of its unauthorized access or change. Systems which handle Top Secret data have a higher security requirement than those which handle data of lower classification. In a security requirements sense, if different levels of data classification exist, the security design problem increases since techniques must be implemented to isolate the different levels, provide the proper degree of security protection, and guard against unintentional or deliberate attempts to gain access to data at unauthorized security levels. The most severe case is that of allowing unclassified data access to uncleared users in a system that contains classified data. Special security precautions must be implemented to insure that classified data is not released to uncleared users either unintentionally or by deliberate intent.

Three levels of security requirements exist depending upon the classification of data in the system and the level of clearance of the user: (1) the data classification is all one level (such as secret) and all users are cleared to that level, (2) different data classification exist (TS, S, C) with users of different clearance levels (TS, S, C), and (3) unclassified data exists with classified data and uncleared users are allowed to access the unclassified data. The security requirements increase as data classification and user clearance levels increase in complexity.

E. SUMMARY

The security design requirements of systems are directly dependent upon (1) user language capability, (2) location and use of terminals, and (3) the level of data classification accessible at a single terminal. Table 2-4 is a summary of these three areas and the descriptions of system characteristics that influence security requirements. The particular security requirements associated with each description are presented and numbered. The numbers are used in later references. In the following sections, systems are categorized and discussed as to their security requirements and security techniques implemented in relationship to these user access capabilities.

TABLE 2-3. SECURITY TECHNIQUES APPLICABLE TO THREAT POSED BY FAILURE ACCESS

Systems	Techniques		
	Hardware	Software	Procedural
407L, NTDS Navy LCC (Prototype)	• Power Control	• Software Diagnostics	Debug Discipline Security Discipline Periodic Maintenance Compartmentalized Maintenance
465L	• Power Control Interrupt • Parity Check • Backup Processor	• Software Diagnostics	
All other systems surveyed	• Power Control Interrupt • Parity Check • Privileged Mode • Priveleged Instructor Set • Memory Bounds	• On-Line Diagnostics • Fail-Soft Backup Modes	
Discussed but not implemented in any system.	• Special Additional Parity Check for Key Circuits • Redundant Key Registers		

TABLE 2-4. SYSTEM DESIGN FEATURES AND RELATED SECURITY REQUIREMENTS

System Design Features	System Security Requirements*
1. User Language	
Off Line	R1
Fixed Transaction	R2-R5
Free Form Query	R6-R12
Assembly and/or POL	R13-R22
2. Terminal Location and Use	
Local only, single need to know	R23-R30
Local only, multiple need to know	R23-R35
Remote/Local, single need to know	R23-R34 R36-R43
Remote/Local, multiple need to know	R23-R34 R36-R43
3. User and Data Clearance Level	
Single Classified	R-44
Multiple Classified	R45-R53
Mixed Classified and Unclassified	R45-R57

*Key to code

R1. The need to identify the user identification with each job
R2. The need to correctly identify terminal identification and relate these to transaction identification.
R3. The need to identify user identification and relate these to transaction identification.
R4. The need to prevent modifications in the linkages established for transactions
R5. The need to certify a user's ability to correctly request transactions.
R6. The need to recognize the classification of data requested in queries.
R7. The need to determine who has inputted a query requisition and from where.
R8. The need to determine the classification of the results file obtained from the query.
R9. The need to recognize repeated attempts to formulate legal queries.
R10. The need to control the distribution of query results files.
R11. The ability to prevent multiple queries from interfering with each others results.
R12. The ability to restrict access to result files to the original creator, or other identified users.
R13. The ability to determine the intent of a program.
R14. The ability to restrict program coding to user mode.
R15. The ability to interpret boundary limits in address statements.
R16. The ability to identify the owners of particular programs.
R17. The need to classify library routines.
R18. The need to limit the capability to update or change library routines on-line.

TABLE 2-4. SYSTEM DESIGN FEATURES AND RELATED SECURITY REQUIREMENTS (Continued)

R19. The ability to identify terminal devices accessed.
R20. The ability to certify that executable segments have not been extended.
R21. The ability to recognize what data is being requested by a program.
R22. The ability to monitor the amount of time any routine is occupying CPU or channel availability.
R23. The ability to secure the terminal area against unlawful entry.
R24. The ability to disconnect terminals.
R25. The need to identify specific terminal addresses.
R26. The ability to mask identification input from terminals.
R27. The ability to determine that the user at an on-line terminal is recognized by the system.
R28. The ability to alert control groups if illegal terminal entries are made.
R29. The ability to trace entered messages to responsible terminals.
R30. The need to validate the format and values of data received from on-line terminals.
R31. The ability to identify individual users at the terminal.
R32. The need to establish the access rights of individual users.
R33. The need to establish the access rights of individual terminals.
R34. The ability to associate given users with given terminals.
R35. The ability to detect attempts to input alternate user identifiers of access words from terminals.
R36. The ability to detect and/or correct noise on transmissions.
R37. The need to secure remote terminal areas from unlawful entry.
R38. The need to prevent easy identification at the terminal.
R39. The ability to control the routing of information.
R40. The ability to monitor the extent of activity at remote terminals.
R41. The ability to encode data so that it cannot be deciphered.
R42. The ability to time-out a terminal and drop it from the line if it has not been used within a specified time period.
R43. The need to recognize the beginning and the end of user sessions at a terminal.
R44. The ability to identify that all users are properly cleared.
R45. The ability to assign classification to files.
R46. The ability to determine the clearance of each user.
R47. The ability to display the classification of hard-copy or screen output.
R48. The need to relate user clearance to data clearance.
R49. The need for a central site for receiving higher classes of data.
R50. The need to identify the clearance level of each terminal.
R51. The ability to assign the highest level of clearance to a composite data request.
R52. The need to upgrade or downgrade any of the clearance profiles.
R53. The ability to assign classifications to individual data elements.
R54. The need to deny requests for classified data to unclassified users.
R55. The ability to assign classification to individual locations and vary it.
R56. The need to determine and log all requests for classified data.
R57. The need to obscure classified residue.

Section 3

CATEGORIZING GOVERNMENT SYSTEMS BY THEIR SECURITY REQUIREMENTS

CATEGORIZING GOVERNMENT SYSTEMS BY THEIR SECURITY REQUIREMENTS

A. INTRODUCTION

This section discusses the characteristics of the seventeen military and three government automated systems surveyed as part of the security technique review.

Certain aspects found to be common to all government systems surveyed are briefly presented. Each system is then assigned to one of the categories of security requirements described in Section 2. The assignment of a system to a category is determined by the system characteristics that describe its user's accessibility. The categories are discussed in terms of the requirements and major techniques implemented by the systems in the categories. The categories to which the TACC may be assigned, depending on its final design features, are treated in greater detail.

B. COMMON ASPECTS OF GOVERNMENT SYSTEMS

All military systems use the common security technique of secure areas at the local computer system center and, where remote terminals were involved, secure areas at classified terminals. No unclassified remote terminals are connected to the classified systems and every system deals with several levels of user and data clearance. Because of this, only the categories that pertain to this system characteristic are shown in Table 3-1, and discussed below. Cryptographic communications techniques are used for all classified remote terminals. (One government system, the FBI NCIC, uses unsecure commercial communication lines connected to remote terminals at authorized law enforcement agencies.) All military systems use the security procedures given in DOD and military service security regulations that cover clearance procedures, security responsibility, and access limitations. Automated security techniques are used in the military systems in addition to these basic common security techniques.

C. SECURITY REQUIREMENTS AND MAJOR TECHNIQUES OF THE SYSTEMS CONSIDERED

The major differences between systems were found to be in their user access characteristics. The security techniques implemented directly relate to user characteristics, and these serve as a method to categorize systems. The three characteristics found to directly affect security requirements are (1) user language capability, (2) terminal location and use, and (3) data and user clearance levels. These three characteristics are explained and discussed in Section 2. Four levels of user language capability, four

levels of terminal location and use, and three levels of data and user clearance influence security requirements. No military system surveyed allowed uncleared users on-line access to a system with classified data, a case that may be necessary in the Post-75 TACC. All systems surveyed deal with multilevel data and user clearance levels. Table 3-1 categorizes the surveyed systems by user accessibility characteristics. The four levels of terminal location and use are shown against four levels of user language capability for the case of mixed data and user clearances. The security requirements increase as the user characteristics change from (1) local only to local and remote terminals, (2) off-line to fixed transactions to free form query to assembly/POL language capability, and (3) common to multiple access rights. The area where the Post-75 TACC could fall is shown by a dotted square.

The systems surveyed fall into eight of the 16 possible categories. Each of these categories are discussed in the following sections in regard to the security requirement, the systems in the category, and the security techniques implemented. For easy reference, the paragraph number where each category is discussed is shown in the lower right corner of each block. Each individual system is described in more detail in Appendix B.

TABLE 3-1. GOVERNMENT SYSTEMS ASSIGNED TO CATEGORIES AT USER ACCESSIBILITY

User Terminal		User Language Capability			
Location	Use	Off-Line	Fixed Transaction	Free Form Query	Assembly/POL Language
Local Only	Common Access Rights		USAF 407L NTDS 1b	STRICOM USN LCC 1c	
	Multiple Access Rights	AFAL RECON FTD NMCS/NIPS AFICCS 1a		NMCS/NOMIS TIPI/NOMIS 1c	
Local and Remote	Common Access Rights		SACCOM 465L TACFIRE TOS Seed Data II FBI NCIC 2a	TACC	AFSS NSA 2c
	Multiple Access Rights			DIA WWMCCS/EXEC 8 Army RAPID 2b	NASA Coherent 2d

1. Local Terminal Systems

a. Off-Line Systems

These systems allow the user access to the system only through manual requests for reports or services. The user is not normally allowed in the local secure computer area and his requests for services are handled by support personnel who interface with the computer hardware, software and data files. The security threats to this type of system are similar to those common to manual systems. The same manual procedural techniques with slight additions to provide for the protection of tape files and direct access memory are used in these systems. Table 3-2 summarizes the security requirements of the systems and the major security techniques implemented for off-line systems. These systems used solely procedural security techniques since their characteristics do not vary significantly from manual systems and there is no need for automated security techniques to protect against deliberate penetration.

b. Fixed Transaction Systems

These systems are similar in characteristics to the previous category except users are on-line to the computer equipment and can enter and access fixed predetermined transactions. The users are at terminals inside the secure area and the transactions that can be accessed at any given terminal are fixed and predetermined. The data associated with each transaction and the processes required to obtain it are precompiled and unchangeable. The security requirement is similar to that of off-line systems except that user and terminal access must be controlled. Automated security techniques were not felt necessary to these systems. Table 3-2 summarizes the security requirements and the major security techniques implemented for fixed transaction systems.

c. Free-Form Query Systems

The ability to format free-form queries on-line adds additional security requirements because the data access requirements of the user cannot be predetermined as in fixed transaction systems. The additional capability of the language also requires techniques to prevent inadvertant or deliberate circumvention of access and process control techniques. Two systems, the Navy LCC and STRICOM, allow access to all files from any terminal by all users; hence, there is no need to implement techniques to limit individual user capabilities. In the other systems of this class, automated techniques are implemented to limit access to specified files by individual user identification. Thus, multiple need-to-know users can use the same terminal to access different classes of data. The two systems that allow multiple access rights at any given terminal (NMCS/NOMIS and TIPI/NOMIS) are versions of the SDC developed ADEPT/TDMS software system. The main security features implemented are user identification codes and user profile tables that identify user rights to files and programs. The user code word also identifies the highest classification of data and programs that the user can access. All programs and files are assigned classification levels and a users' access is prohibited unless his clearance level is equal or higher than that of the data file and his profile grants the right to access the data. (See also the discussion of ADEPT/TDMS in Appendix B).

TABLE 3-2. LOCAL TERMINAL, OFF-LINE AND FIXED TRANSACTION SYSTEMS

System	Security Characteristics	Major Techniques		
		Hardware	Software	Procedural
AFAL Recon FTD NMCS/NIPS* AFICCS	Batch System, Off-Line TS, S, CONF DATA Special Categories Data Analysis and Summary Reporting	IBM 360 System Parity Check Circuits Memory Bounds (BAR) Privileged Mode plus Instruction Set	None	Standard Security Procedures as in Manual Systems Secure Area
USAF 407L NTDS	On-Line Fixed Transactions S, CONF DATA Air Situation Monitoring plus Aircraft Vectoring	Power Failure Interrupt	None	Standard Security Procedures as in Manual Systems Secure Area

*NMCS/NIPS limits support personnel access to files by a card Identification Input Check on Support User Rights

All four systems are local terminal only and secure area procedures are used to limit terminal access to cleared user personnel. They are all in the development and test stage and represent the first category of systems that implement automated security techniques for security control. Table 3-3 summarizes the security requirements and the major security techniques implemented for each of the four systems.

2. Remote Terminal Systems

All remote terminal systems are summarized in Table 3-4. The categories of systems are described below.

a. Fixed Transaction

Five systems surveyed fall into this category, four military and one government (FBI). The remote terminal characteristics add additional requirements for techniques to protect against communications line penetrations, remote terminal area protection, and user identification. The four military systems use dedicated communications lines with cryptographic techniques to prevent access to data by wiretapping or electromagnetic radiation. The FBI system uses common carrier lines with no encoding of data. Remote terminal secure area procedures are used by all the military systems. The FBI remote terminals are located in law enforcement offices throughout the country.

Preconceived fixed format transactions with common access rights at a given terminal allow implementation of access control by specifying only which transactions are allowed at a given terminal and the data files the transaction can access or change. All the systems in this class have implemented this level of security access control. In addition, the SEEK DATA II system has implemented a code word identification check for terminal sign-on in the system. Since these systems fall into categories applicable to the Post 75 TACC, their system characteristics and security techniques implemented will be discussed in more detail in the following paragraphs.

(1) SAC Command and Control System (465L)

The security requirements of the Strategic Air Command and Control System include the handling the Top Secret, Secret, Confidential, and special data classes. The data is input from on-line remote terminals with common access rights and consists of situation reporting. The data is used for analyses and summary reporting accomplished at local fixed-format on-line terminals. Hardware techniques applicable to security requirements include crypto secure communications lines and a back-up computer to be used in case of hardware/software failure. No automated software technique directly applicable to security is implemented but the system allows only preconceived fixed format input/output at a given terminal. Procedural techniques include the control of access to secure areas at the computer center and at all remote on-line terminals. Standard security procedures in compliance with Air Force security regulations are used within the secure areas.

TABLE 3-3. LOCAL TERMINAL, FREE-FORM QUERY SYSTEMS

System	Security Characteristics	Major Techniques	
		Hardware	Software
STRICOM	On-Line, Free-Form Query Common Access Rights TS, S, CONF Data Special Categories Data Analysis and Summary Reporting	IBM 360/50	None
USN - LCC	On-Line, Free-Form Query Common Access Rights S, CONF Data Navy Force Logistic Planning Analysis and Summary Reporting	AN/USQ-20 Power Failure Interrupt	File and Data Element Access Terminal Profile Table Read/Write Control
NMCS/NOMIS TIPI/NOMIS	On-Line, Free-Form Query Multiple Access Rights TS, S, CONF Data Special Categories Data Analysis and Summary Reporting	IBM 360/50 System	File Access Control User Profile Table Write/Read/Protect Consecutive Password List

(2) Army Tactical Operational System (TOS)

This system is a commercial test bed used to test automation concepts for command and control of the field army. The security requirements include the handling of Secret, Confidential, and special data classes. Remote terminals are used for situation reporting, analysis, and summary reporting. The system is a message driven, table lookup design that provides for preconceived fixed format input/output at given terminals. Access to data and programs is controlled by terminal identification, hence, terminal common access rights are used and there is no requirement to identify individual users.

The major hardware security techniques used are cryptographic secure communications lines and the hardware multiprocessing features of a third generation computer system, the CDC 3300. The CDC 3300 has a processor mode register with privileged instruction set for the executive mode, and core memory bounds (paging). These features isolate the execution of individual programs and control all input/output through the executive software. It provides protection from inadvertant or deliberate attempts to access programs or data files other than those specified by the executive programs. Data parity checks are used in the core memory and between core memory, CPU, and IOCP to detect hardware errors.

The software security techniques consist of transaction access rights at the file level for data. A terminal profile table is maintained that specifies the transactions which are authorized at a given terminal and the right to read or write in a given file by a given terminal.

(3) Seek Data II

This system is a first step automation of the TACC using commercially available computer equipment. It is now being implemented in South East Asia. The security requirements include handling of Top Secret, Secret, Confidential, and special data classes. Remote terminal, on-line and off-line are used for situation reporting and air lift frag order dissemination. At the TACC, local on-line terminals are used for air lift frag order preparation and summary reporting. The system is planned to be expanded to include strike and recon planning, and near real-time situation monitoring. Preconceived fixed format input/output at given terminals is used and access to data and programs is controlled by terminal identification. Terminal users are required to log in with a code word to the system when they first start to use the terminal. Once they have successfully "logged in," they are allowed access to all transactions authorized to that particular terminal.

The major hardware security techniques used are cryptographic secure communications lines and the hardware multiprocessing features of a third generation commercial computer system, the IBM 360/50. The IBM 360/50 has a processor mode register with privileged instruction set for the executive mode and core memory bounds (base address register). It includes parity checking in the core memory module and between core, CPU, and I/O channels. It provides protection of the same class as the COC 3300 used in the TOS system.

The software security features include a code word check to identify system users and terminal identification for access to fixed transactions.

(4) Army TACFIRE System

This system is the automation of the Army tactical artillery fire control. Its purpose is to receive fire requests, control the technical fire, and report battle damage. The security requirements include the handling of Secret, Confidential, and special data. The system has one unique security requirement, the need to ensure the validity of data received from a forward observer on-line remote terminal that must be highly mobile and is subject to enemy seizure.

The hardware security features are those associated with the Litton 3050 military computer and include privileged mode, privileged instruction set, memory protect, and parity between CPU, core and IOCP. Cryptographic secure communications lines are used except for the input channel from the forward observer. The system will accept messages from the forward observer and send back an acknowledge signal; however, no data is passed from the system to the forward observer on this unsecure communications line.

The software security techniques include a consecutive password check of code word inputs from the forward observer before data input is accepted into the system. Other remote terminal users are allowed authorized preconceived transactions by terminal identification and are connected through cryptographic secure communications lines.

(5) FBI NCIC Systems

This system provides national criminal information on-line to law enforcement agencies throughout the country. The system can be queried for criminal records, gun records, stolen property records, stolen car records, and for Federal want warrant records. Remote terminals are connected through common carrier lines to a central commercial computer processing center. There is no requirement for handling classified data in this non-military system but there is a privacy requirement for protection of information in the files.

Hardware privacy techniques are, again, those associated with third generation computer equipment; in this case the IBM 360/65. No special hardware or software security techniques are used in the system except that the system does check a transaction request against that authorized for the requesting terminal.

b. Free-Form Query

Three of the surveyed systems fall into this category, two in use (DIA and Army RAPID) and one proposed for implementation to meet the World Wide Military Command and Control System Specification (WWMCCS/EXEC 8). The characteristic of this category that differs from the remote terminal, fixed transaction, common access rights system category is that users with different need-to-know are allowed use of the same terminal. Therefore, the system must include techniques that identify unique users so that their program and data accesses can be compared with known access rights to prevent release of data not in their need-to-know class. This category of system also

allows free-form query language capability to the user so data to be released to the terminal cannot be predetermined. The language capability provides more capability for unintentional or deliberate penetration of security techniques. These systems have implemented or proposed more security techniques directly applicable to the TACC than any of the other military systems surveyed.

(1) WWMCCS/EXEC 8 System

This is a system proposed by UNIVAC to meet the requirements of the WWMCCS Specification. It is a modification of the Executive 8 software system that operates on the UNIVAC 1108 computer system. The security requirements of the WWMCCS Specification call for the handling of all data classification levels except unclassified, and special data classes. The system specifies local and remote on-line terminals with cryptographic secure communications lines connecting to the remote terminals. Users with different need-to-know rights can use the same terminal.

The applicable hardware security techniques again are those normally associated with third generation computer systems (executive mode, privileged instruction set, memory protect (lock and key), and parity checking). No special additional hardware features were proposed for additional security protection.

The software security features include the use of user passwords. These passwords are transformed internally to prevent access by other system users. User profiles specify the programs and files the user can read/write/execute. In addition, before storage is allocated for a user task, the storage is cleared to prevent any readout or dump of previous programs or data in the area.

(2) DIA System

This system is an operational system using commercially available computer equipment (GE 635) with remote and local on-line terminals. The security requirements of the system include the handling of Top Secret, Secret and Confidential data and special classes of data. The data in the system is used for intelligence analysis and summary reporting. Users with different need-to-know use the same terminal and have the capability to input free form query requests. A computational and query language are both available for use by the terminal operators.

The hardware techniques applicable to security are those associated with a third generation machine. The GE635 has privileged mode, privileged instruction set, memory bounds (base address register), and parity checking between core, CPU, and IOCP. A prototype output security check/number of characters register was tested in the system but was not used because experience revealed no misrouting of data on output channels. The register checked for the correct channel number for data output and allowed a specified number of characters to be sent on the channels. No more characters were sent until the channel number was again compared by the CPU and the register reset. Cryptographic secure lines are used to connect to remote terminals.

Software security features implemented included the use of user passwords, user profile tables that specify file access, program access, and read/write/execute rights for each user. The executive system in use (the Dartmouth System) was modified to insure that memory bounds and privileged mode control could not be circumvented by users. A unique security monitor program was implemented. Operating in the executive mode it tests on-line the hardware and software security features of the system by deliberately issuing unauthorized calls and checking to see that the system refuses data or program access.

(3) Army Personnel Information System/RAPID

This system provides on-line processing of Army personnel actions, as well as unit reporting, analysis, and summary reporting of personnel actions. The security requirements of the system include the handling of Secret and Confidential data at remote on-line terminals. Users with different need-to-know use the same terminal and are provided with a free-form query capability.

The applicable hardware techniques are those associated with the CDC 3300, the same as the Army TOS Test Bed System. Cryptographic techniques are used to secure the communications lines that connect the on-line remote terminals.

The software security techniques implemented include the use of passwords, terminal profiles, and user profile tables that include the authority to read/write/execute data and programs. Access to data is controlled at the data element level instead of the file level. A log is maintained of all data element accesses, including the time, terminal ID, user ID, and full text of the entry.

c. Assembly/POL Language, Common Rights Systems

The two systems in this category (NSA and AFSS) allow users at remote terminals to enter programs in assembly or POL language. This level of language capability implies that special security precautions must be implemented to prevent circumvention of access control and process control techniques, either inadvertently or by deliberate intent. Both systems control access to programs or files by terminal rights. Both remote and local terminals are in secure areas and only personnel with the authority to use the terminal are allowed access.

Hardware features of both systems applicable to security are those furnished by the UNIVAC 494 computer system for multiprocessing control. They include privileged mode, privileged instruction set, memory bounds (base address register) and parity checks between core, CPU, and IOCP. Cryptographic secure communication lines are used to connect remote terminals.

The software security features include the implementation of a special executive that protects against circumvention of access control, process control, and memory bounds features. Access to data and programs is controlled by terminal profile tables that authorize terminals to read/write/execute data. Access to data is controlled at the file level.

d. <u>Assembly/POL Language, Multiple Access Systems</u>

One proposed government system, the NASA Coherent system falls into this category. The system has beyond that of the previous class the additional security requirement of allowing users with different data requirements and need-to-know to use the same terminal.

The hardware features of this system have not been selected. The software security techniques proposed include the use of User Profile Tables that specify the user rights to programs and to data elements within files. Security logs of all data element accesses and an on-line security monitor program are proposed. The security monitor program is limited to allowing a given number of access attempts before alarming system support personnel.

D. SUMMARY

The security requirements and techniques implemented increase as remote on-line terminals are added and as the language capability of the user increases. The requirement to process more than one need-to-know at a given terminal also requires additional security techniques to identify the terminal user. The Post 75 TACC falls into the class of remote terminal system with either user fixed transaction or free form query language capability. Surveyed systems that fall into this category are the following: 1) common access rights: 465L, TACFIRE, Army TOS, Seek Data II and 2) multiple access rights: WWMCCS/EXEC 8, DIA, Army Personnel System/RAPID.

TABLE 3-4. REMOTE TERMINAL SYSTEMS

System	Security Characteristics	Major Techniques	
		Hardware	Software
SAC COM (465L)	On-Line Fixed Transactions Common Access Rights TS, S, C Data Special Categories Data	Cryptographic Secure Comm Duel Computers	File Level Access by Transaction
Army TACFIRE	On-Line Fixed Transactions Common Access Rights S, C, Data Artillary Fire Control Reporting, Fire Direction	Cryptographic Secure Comm One Unsecure Comm Input Line (Forward Observer) Litton 3050 Computer System Parity Check Circuits Memory Bounds Privileged Mode Plus Instruction Set	Sequence Code Word Input and Verification from Forward Observer File Level Access by Transaction
Army TOS (Test Bed)	On-Line, Fixed Transactions Common Access Rights S, C, Special Category Data Field Army Command Plus Control Reporting, Analysis Summary	Cryptographic Secure Comm CDC 3300/1700 System Parity Check Circuits Memory Bounds (Paging) Privileged Mode Plus Instruction Set	File Level Access by Transactions Terminal Profile Table Read/Write Control
Seek Data II	On-Line Fixed Transactions Common Access Rights TS, S, C, Special Category Data Situation Reporting, Airlift Frag Order Preparations, Summary Reporting	Cryptographic Secure Comm IBM 360/50 System	System Access by Code Word File Level Access by Transaction
FBI NCIC .	On-Line Fixed Transactions Common Access Rights Privacy Data Crime Data Query	Commercial Comm Lines IBM 360/65	File Level Access by Transactions

TABLE 3-4. REMOTE TERMINAL SYSTEMS (Continued)

System	Security Characteristics	Major Techniques	
		Hardware	Software
WWMCCS/EXEC Eight	WWMCCS Specification On Line, Free-Form Query Multiple Access Rights TS, S, C, Special Category Data Reporting, Analysis, Summary	Cryptographic Secure Comm UNIVAC 1108 System Parity Check Circuits Memory Bounds (Lock and Key) Privileged Mode Plus Instruction Set	File Level Access User Password/Transform Internally EXEC Access Rights User File Plus Program Profiles Storage Overwrite Profiles
DIA	On Line, Free-Form Query Multiple Access Rights TS, S, C, Special Category Data Analysis and Summary	Cryptographic Secure Comm GE 635 System Parity Check Circuits Memory Bounds (BAR) Privileged Mode Plus Instruction Set	File Level Access User Profile/Codewords On-Line Security Monitor Read/Write/Execute Control
Army Personnel System/RAPID	On Line, Free Form Query Multiple Access Rights S, C, Data Reporting, Analysis, Summary	Cryptographic Secure Comm CDC 3300 System	File/Data Element Access User Profile/Codewords Read/Write/Execute Control
AFSS NSA	On Line, Assemble/POL Multiple Access Rights TS, S, C, Special Category Data Analysis and Summary Reports	Cryptographic Secure Comm UNIVAC 494 System Parity Check Circuits Privileged Modes Plus Instruction Set	File Level Access Special EXEC Design Terminal Profile Read/Write/Execute Control
NASA Coherent	On Line, Assemble/POL Multiple Access Rights Special Data Categories Information Processing Utility	Not Decided	User Profiles Data Element Access Security Logs Security Monitor

Section 4

CATEGORIZING COMMERCIAL SYSTEMS BY THEIR SECURITY REQUIREMENTS

Section 4

CATEGORIZING COMMERCIAL SYSTEMS BY THEIR SECURITY REQUIREMENTS

A. INTRODUCTION

In this section, the security requirements of the commercial systems surveyed are presented. The systems are categorized into the same matrix defined in Section 2 and used for the government systems considered in Section 3, i.e., according to the degree of accessibility afforded by the combination of terminal location and user language capabilities. The major techniques implemented in each category of systems are presented briefly, including a discussion of techniques to protect against unintentional compromise. The systems by category are presented in Table 4-1 (the government systems are included as well for comparative purposes) and discussed below. Certain techniques common to most of the systems considered (labelled "major operating systems" in Table 4-1) and developed to satisfy requirements for process control while multiprogramming are treated as a single separate category.

B. COMMON ASPECTS OF COMMERCIAL SYSTEMS

In a multiprogramming environment (or multiprocessing in certain cases), there are several critical functions normally performed by the executive software routine that are applicable to satisfying system security requirements. These are implemented in every case to permit the system to successfully manage an environment of competing requests for a limited set of resources from concurrently running programs. However, because they concern themselves with insuring the integrity of a processing sequence, they can be considered, and have become in many cases, the only security techniques for protection against unintentional compromise.

The routines which control the user terminal input/output maintain polling lists of legal terminal addresses, have flag-bits set to indicate the current status of terminals, accept an acknowledge code, or a hardwired answerback code from terminals, can disconnect idle terminals, and usually perform message parity and longitudinal redundancy checks to validate that the message content received is properly coded. The routines request the user to log in, usually by identifying the job to be run, but frequently also by providing some type of charge number or identification that can be assigned for accountability purposes. In addition, the point-of-entry, perhaps the time-of-entry, a priority, specific resources, and some kind of internal identifier are also associated with the user. Frequently the received messages are logged on secondary storage for audit-trial creation and statistical analysis.

TABLE 4-1. COMMERCIAL SYSTEMS (UNDERLINED) ASSIGNED TO CATEGORIES OF USER ACCESSIBILITY

User Terminal		User Language Capability			
Location	Usage	None	Fixed Transactions	Free-Form Query	Assembly/POL Language
Local Only	Common Access Rights	(Major Operating Systems)	USAF 407L NTDS	STRICOM Navy LCC <u>AESOP</u>	All Systems
	Different Access Rights	AFAL RECON FTD NMCS/NIPS AFICCS	<u>ESMIS</u>	NMCS/NOMIS TIPI/NOMIS <u>GIM</u> <u>GIS</u> <u>DM-1</u>	All Systems
Remote and Local	Common Access Rights	(Major Operating Systems)	TACFIRE TOS SAC COM 465L SEEK DATA II FBI NCIC		AFSS NSA
	Different Access Rights	<u>TSOS</u> <u>MARK II</u>	<u>AAS</u>	DIA WWMCCS/ EXEC 8 <u>MARS</u> ARMY RAPID <u>MIS</u>	NASA COHERENT <u>MULTICS</u> <u>BTSS</u> <u>ALCOM</u> <u>ADEPT/TDMS</u>

TACC

When allocating time, space, and peripherals to concurrently processing tasks, the executive must maintain lists indicating task status, resource status, occupied and available units of allocation, etc. These lists, particularly those concerned with partitioning memory among several resident tasks, are usually created and maintained using privileged executive instructions. There are complimentary sets of either boundary address registers, page registers, segment state indicators, lock and key registers, or execution status (read and write protect) flags that can be set, loaded or read only by privileged instructions. A task can be effectively isolated then to a defined set of resources; any attempt by the task to process outside the boundaries of this set is usually treated as a system fault, suspending the process of the task and trapping to executive diagnostic routines to determine the cause and possible solution of the errors.

The problem is complicated by the use of common buffer areas, shared or re-entrant use of executive service routines, and common data bases. However, techniques have been implemented using pop-up/pop-down list processing hardware and software pointers and status flags to maintain the integrity of processing threads, and to ensure that the linkages between parent tasks, sub-tasks, service routines, and data buffers are consistently maintained. However, the more these techniques try to maximize concurrent usage without "locking out" or dedicating resources to a singe task, the more complex, time-consuming and overhead prone they become.

Insofar as handling data files and peripherals are concerned, few systems will accept direct Input/Output commands from tasks, but instead trap to and rely on executive I/O modules ranging in complexity from individually allocated device drivers to multi-user data management systems. In most systems, data and processes are not considered to be "owned" by any individual user, but are usually allocated to a single user at a time. However, in some cases ownership lists for data files can be created, and processes can be assigned to either private user or public system libraries. In a few systems, concurrent use of data files is permitted.

The problem of reliability and recovery is generally addressed by either redundant parallel devices and/or processing, or by maintenance of an audit-trail in conjunction with backup program and data files. Fail-safe is the goal, achieved to a greater or lesser degree depending on how one regards recovery time; fail-soft has been attempted in a few primarily communication oriented systems.

There are many hardware and software techniques currently and commonly in use on third-generation systems that have direct or indirect application to security requirements. Since these routines were not implemented with security as a design goal, but with processing efficiency and multiprogram integrity instead, they do not constitute effective barriers against deliberate penetration attempts. A large part of the problem is due to the dearth of effective complimentary administrative and procedural techniques. However, they provide a relatively secure means of protecting against compromise from unintentional release of data, and raise the level of effort required to deliberately compromise data above that experienced in typical manual systems.

C. SECURITY REQUIREMENTS AND MAJOR TECHNIQUES OF THE SYSTEMS CONSIDERED

1. Local Terminal Systems

a. Batch Input Systems

The security requirements of batch input systems are primarily those that ensure a secure area, and as such are similar to manual systems, since the user does not have direct access capability to any of the shared system components. In both the commercial systems in this category, the RCA Time-Shared Operating System and GE GECOS/MARK II Time-Shared System, most of the software protection features discussed above as common multiprogramming techniques were implemented. In addition, file ownership can be declared. No consideration is given to the security requirements of remote terminals in the RCA systems. The GE system provides hardwired answerback code for every on-line terminal. The systems are summarized in Table 4-2.

b. Fixed Transaction Input

The security requirements for fixed transaction input systems are described in Section 3, paragraph C.1.b. The single commercial system fitting the characteristics of this category is the Employment Security Management Information System (ESMIS), shown in Table 4-3. ESMIS is a data management system developed by Auerbach Corp. for the Federal Labor Bureau and the states of Michigan, Utah, and Florida. ESMIS is a variant of the DM-1 software system designed to operate on the 360-50. The original approach to providing for the privacy of the data base was to create a terminal profile. However, since various types of users had access to the same terminal, it became apparent that any sophisticated user could browse through the entire data base without restriction. Consequently, a user profile/data element profile approach was adopted. In addition, multiple users were permitted concurrent access to the ESMIS data base. An effective but costly block lockout scheme was implemented to counter the potential problem of mutual interference of simultaneous data base updates by more than one user.

c. Free-Form Query Systems

The security requirements for free-form query systems were described in Section 3, paragraph C.1.c. One of the systems reviewed, AESOP, assumes that all users at any terminal have essentially the same need to know. Consequently, no techniques were implemented to specifically identify users. File-level access is established and security headers and trailers are printed on all displayed and hard-copy output. Since AESOP was designed to test software concepts and not specifically as an operational system the security requirements were not considered to be paramount. There are three systems that permit users with different need-to-know to access the data base from the same terminal. They are all general-purpose data management systems, and rely on the operating system/hardware for terminal protection mechanisms. Each of the three has implemented some variant of a scheme to assign separate

access codes at the data element level. GIS and DM-1 use the user's identification to determine his classification level. GIM maintains a full user/terminal profile. Table 4-3 summarizes the security characteristics and major techniques implemented by the systems in this category.

2. Remote Terminal Systems

a. Fixed Transactions

The security requirements for fixed transactions systems are similar to those that might be required by the Post-75 TACC design. Access is permitted through the same terminals to users having different need-to-know rights. A scheme is required to identify the remote terminal user and his need-to-know. Protection is required for communication transmission and to prevent illicit terminal access. Since only fixed-transaction inputs are accepted from any terminal, a predetermined relationship between data, processes, and terminals exists.

The IBM 0S/360/AAS System security characteristics and major techniques are summarized in Table 4-4. This system was developed to provide on-line information services to all branch sales offices. There are over 300 such offices in the country. The services include branch office administration, order entry and status, competitive evaluation, and sales prospect profiles. The users of the system range from regional marketing managers to demonstration personnel. More than 1700 teletype and display terminals are connected on-line through ten 360/30's to 18 billion characters of data accessed by a sextuplexed 360/65 system. More than 800 man-years of development effort have been expended on the system thus far.

Since even small increments in thruput, storage, or response time can have large effects in a system of this size, IBM carefully considered what security techniques should be implemented to provide the desired level of protection. The coordination between complementary hardware, software and procedural techniques was emphasized in all design phases. Some of the techniques include:

(1) Physical Security, such as locking doors, files, cash boxes and supply rooms;

(2) Separation of duties, including the involvement of two or more people in sensitive transactions such as entering an order;

(3) Accountability, including making a person responsible for all of the entries that he makes to the system;

(4) Audit, including the ability to determine who did what, when and under what circumstances; and

(5) Supervision, including recognizing a variance from normal practice, procedures, or standards and taking prompt corrective action.

The hardware techniques include voice identifiers at remote sites, character-blanking on CRT-terminals, hard-wired terminal identification codes, dedicated common-carrier lines, and multi-processing extensions to the 360/65.

TABLE 4-2. COMMERCIAL SYSTEMS, REMOTE BATCH INPUT

System	Security Characteristics	Major Techniques	
		Hardware	Software
RCA TSOS	Multiple Users, Public and Private Access Remote Multifunction Terminals Batched Assembly/POL Submission Time Sharing Services	RCA 70/46	File Level Access Rd/Wr/Execute Mode
GE GECOS/ MARK II	(As Above)	GE 635	File Level Access User/Account ID Rd/Wr/Execute Mode

TABLE 4-3. COMMERCIAL SYSTEMS, LOCAL TERMINAL ACCESS

System	Security Characteristics	Major Techniques	
		Hardware	Software
IBM OS 360/ESMIS	Local Terminals, Different Access Private Access, Several levels Fixed Transactions Employment Matching	360/50	Data Element Profile User Identification Terminal Identification Block-Level Lockout
MITRE COGENT/ AESOP	Multiple Users Local Terminal, Common Access Rights Public, Private Data General Purpose	IBM 7030 (Stretch) Parity	File Level Access Security Headers
TRW ITDS/ GIM	Multiple Users Local Terminals, Different Access S, CONF, Public, Private, Special General Purpose System	360/40 and Up	Data Element Access User Identification User Profile Terminal Identification Security Headers Special Access Verbs
IBM OS 360/GIS	Batch System, Multi-User Terminal Queries, Local Private Data Query Extraction and Report Formulation	360/50 and Up	User Log in-Code Data Element Access Code
GE GECOS/ DM-1	Multiple Users Freeform Query Public, Private, Special Data General Purpose System	GE 635	Intermediate Files Data Element Access Level

The components of the AAS software security system are the Identification, Authorization, and Audit Trail subsystems.

(1) Identification Subsystem

An AAS user is identified by his IBM Employee Serial Number. To indicate that he is the person identified by a given number, the user must also enter a four-letter Security Code. This code is unique to the user, and is known only to him and the system. He must enter the code once for each transaction he wishes to perform. The code is generated automatically by the system. It is changed monthly to protect against an undetected compromise or upon demand in the event of a detected compromise. The code is mailed to the user in a secure envelope.

(2) Authorization Subsystem

A user must be authorized by his management to perform a transaction. The system maintains a list of the authorized transactions. Management is provided with a real-time transaction – "Security Authorization" – for creating, adding to or deleting from this list. The system maintains authorization rights as a user-profile file, currently in excess of 15,000 entries, containing name, employee number, location, and list of legal and capable transactions. For some transactions a user, in addition to being authorized by his own management, must meet an AAS prerequisite. The prerequisite is normally the completion of a Computer Assisted Instruction (CAI) course which covers the transaction. When a user meets the prerequisite for a transaction, it is recorded in his capability profile. This is normally handled automatically by the system; however, it can also be entered by the security administrator. Any two consecutive entries by the same user at the same terminal causes the terminal to be locked. It can only be opened by a special code assigned to the Security Officer.

(3) Audit Trail Subsystem

The audit trail in the AAS system is provided by the "Data-File Journal". This file contains one entry for each change to the data base and one for each transaction. The entry for a change to the data base identifies the terminal, the transaction, and the transaction record. The transaction record identifies the user. Thus, every change to the data base can be traced back to the person responsible. In addition to the Data-File Journal, a record is kept of all Security Violations (Security Violation Log), and of all changes to the user profiles.

b. Free Form Query

The security requirements for free form query systems were described in Section 3, paragraph C.2.b. Two commercial systems are assigned to this category; the CDC MASTER/MARS system and the IBM MIS system. Both are summarized in terms of security characteristics and major techniques implemented in Table 4-4.

TABLE 4-4. COMMERICAL SYSTEMS, SECURITY REQUIREMENTS SIMILAR TO THE POST-1975 AUTOMATED TACC

System	Security Characteristics	Major Techniques	
		Hardware	Software
IBM OS 360/AAS	Remote Terminals, different access privileges Public, Private, Special categories Fixed Transactions Order Status, Branch Admin. Competitive Evaluation and Prospect Lists	360/65 Terminal Answer-back Terminal Control Code Suppression Voice Identification	User Profile Element Profile Log-in Dialogue Randomized Access Codes On-Line Security Monitor CAI Ranking
CDC MASTER/ MARS	Remote Multi-function Terminals Freeform Queries Public, Private, Special Data General Purpose System	CDC 3300	User Profile Terminal Profile Element Rd/Wr Privilege words Element Need to Know Flags Report Combinations
IBM OS 360/MIS	On-Line System, Multi-user Remote terminals, different privileges Special Categories, Public, Private Administrative Data Management	360/50 Terminal Control Code suppression Terminal Abort Time	Data Element Access Code User Identification User File Profile Special Access Values

(1) MASTER/MARS

The MASTER/MARS system is a generalized on-line information handling system built for the 33/3500 hardware and designed to accommodate many users interacting with a common data base. The hardware was already described in the discussion of the Army TOS. The MASTER Operating System provides all of the protection mechanisms commonly associated with third-generation systems to protect against inadvertent compromise. MARS provides techniques for user identification, terminal identification, data element classification, logging, and system recovery. The problem of remote terminal communication transmission is handled by the terminal identification technique. Every user is identified by an assigned User Name and User Password. These are input each time the user logs in, and provide the key to the user profile table.

MARS allows the ranking of data elements up to 64 levels; level 0 indicates no sensitivity and level 63 indicates the highest sensitivity. Each user allowed access to the data base is assigned an element sensitivity level for each data file available to him. This information is kept in the Data Privacy File (DPF). When each data element of a data file is defined, an element sensitivity level is associated with the element in the File Definition File (FDEF). This level is compared against the user's sensitivity level for the file. The user's level must be equal to or greater than the element level before that element is available to him. Each MARS user is assigned

0-18 flags for each data file available to him. These flag assignments are also kept in the Data Privacy File (DPF). The 18 flags are internally tested by MARS programs and are collectively referred to as a data privacy mask. MARS allows up to 18 classes of data elements for each data file. Each class has its own need-to-know flag. Any number of data elements can belong to the same class. Also, one data element can belong to all classes. When a data element is defined, a need-to-know flag can be associated with that element. This optional flag association is in addition to the element's sensitivity level. Both element sensitivity level and need-to-know flags are created when a given data file is defined. Element sensitivity level and need-to-know flag information are physically stored together in a word called the data element privilege word (DEPW). There is one DEPW in MARS for every valid user/data file combination, defined data element for every data file, remote terminal or class of terminals, and remote-terminal-destined results file. MARS data files are protected in three ways. Only those people with proper authorization are allowed to change the definition of a data file (e.g., delete a whole file or an index of the file), change the contents of a data file, or retrieve information from a data file. The MARS user will generally be referencing many data elements at a time. To create a query results file, he specifies elements in his retrieval criteria and elements in his output statement. MARS uses the DEPW of each data element referenced to form a composite DEPW. This DEPW reflects the sensitivity of the results file. A composite DEPW consists of the highest level and all the flags from the contributing DEPWs.

(2) IBM MIS

MIS is a general data management system that has been implemented in several multi-user on-line systems. It is designed to run on the 360/65 under OS-360 and uses the features of both these systems to protect against inadvertent compromise. Additional hardware techniques implemented are terminal character suppression and an optional terminal time-out that will automatically disconnect an idle terminal. The Software Security Module prevents access to sensitive data without a previously established "need-to-know" and approval for access. Security control is coordinated by entries in the Field Description Table, the User Description Table, and the Access Description Table. Every field or data area is assigned two codes for security purposes, which are entered in the Field Description Table. The first is a category code, which separates all the fields of data in the system into groups. The second code establishes the user clearance level within each category. The User Description Table can include five separate identifiers for every user or user access can be defined by complete access to specific files. The five identifiers are user name and number, user address, inquiry code and type, files and level of access, and specific values of named fields. An Access Description Table is used for limited or special access, when a degree of security clearance is required. It is necessary only when the corresponding access description number has been entered in the User Description Table, indicating that the user has some clearance to the indicated file. The Access Description Table is then referenced to determine what data the user is cleared for. All level and access checks are made when an inquiry is received and prior to any data retrievals.

c. Assembly/POL Language Capability

The security requirements for assembly/POL language capability systems were discussed in Section 3, paragraphs C.2.c and d. Four commercial systems are assigned to this category and are summarized in terms of security requirements and major techniques in Table 4-5. The most advanced security designs encountered in any of the systems reviewed, with the exception of NSA and DIA, were implemented in these systems, primarily to provide privacy for and against the sophisticated users that were expected to be interacting with them. In two cases, MULTICS and ALCOM, the procedural costs of administering the system were so great that most users chose to ignore much of the optional protection mechanism. One of the systems, ADEPT/TDMS, is being used in two military test-beds (TIPI and NMCS/NOMIS).

TABLE 4-5. COMMERCIAL SYSTEMS, REMOTE TERMINALS WITH ASSEMBLY/POL LANGUAGE CAPABILITY

		Major Techniques	
System	Security Characteristics	Hardware	Software
Project MAC/ MULTICS	Remote Multi-function Terminals Assembly Language Program Input Public, Private, Special Data Information Processing Utility for Multiple Concurrent Users	GE 645 PIS Memory Bounds (Page Registers) Segment Pointers Parity Ring Descriptor Bits	User Profile Account & User ID Nodal Access Control Ring Brackets for Processors & Data
BTSS	(Same as above)	GE 635	File Level Access Control User ID/Account ID Rd/Wr/Execute Mode
ALCOM	(Same as above)	PDP-10 PIS Memory Bounds (Paging) (Lock & Key) Parity Terminal Lockout Answer-back Drum	File Level Access Control User/Project/Co ID (Transformed to internal) Rd/Wr/Execute Mode Relocatable Exec File Transform (Encoding) System Log
SDC ADEPT/ TDMS	(Same as above)	IBM 360/50 Also, Page Lock & Keys Fetch Protection	User ID/Terminal ID/ User Profile File Level Access Log in Dialogue Residue Removal Security Audit

SECTION 5

CATEGORIZING AND DESCRIBING SECURITY TECHNIQUES

Section 5

CATEGORIZING AND DESCRIBING SECURITY TECHNIQUES

A. INTRODUCTION

In the preceeding sections, the systems surveyed during the data collection phase of the study were categorized in terms of their security requirements. The characteristics of each category were described, and the major techniques implemented by the various systems assigned to each category were presented.

In this section all of the hardware and software and the major procedural techniques encountered are described and categorized. Three different methods of categorizing the techniques are used. The first two describe the individual techniques and place them in perspective with each other and the general threat to which they apply. The third is the same as that used to categorize systems, and develops the sets of techniques that are applicable to each category of system characteristics.

B. CATEGORIZING TECHNIQUES BY SYSTEM PARTS

The 95 techniques encountered in the various systems and studies surveyed are described individually in this section. The software techniques are assigned to categories that correspond to the major functional routines of an on-line system, that is the user interface, the terminal subsystem, the executive (or monitor or operating system), and the file handler. An additional category containing techniques not conveniently assigned to any other group is added. The hardware techniques are assigned to categories that correspond to the major devices and components in an on-line system, that is, the Central Processor, Main Memory, Input/Output Controller, Direct Assess Memory (Secondary Storage) Controller, and Terminals. A similar category of convenience is also used. The procedural techniques are assigned to categories that correspond to the four major functional requirements of a security scheme, that is, classifying and declassifying material, safeguarding data, accounting for classified material, and disseminating classified material. A common coding scheme is used for all techniques. It consists of a two or three digit code. The first digit indicates whether the technique is Software (S), Hardware (H), or Procedural (P). The next digit(s) sequentially number the techniques as they are addressed by functional part.

C. DESCRIPTION OF SECURITY TECHNIQUES

1. Software Techniques

The Software Techniques encountered are described below, numbered in accordance with the general coding scheme. All of the techniques described are listed by

functional area in Table 5-C-1. Those techniques that have been discussed but not implemented are indicated with an asterisk in the table.

a. User Interface

The user interface is the point at which the user becomes known to and interacts with the system. In a secure system, only known users can be permitted access. The proper identification of the user is necessary for accountability and, in a system that allows multiple need-to-know access at a terminal, to determine the access rights to be associated with the task that is initiated by a user's input request.

S1. USER SECURITY CLEARANCE

The user security clearance is the assignment to each user of a code word indicating the highest classification level of data to which he has been authorized access. It presumes that the data at least is also classified in the same way, although it is applied in its simplest form to distinguishing systems programmers from end-users. Generally, the code word consists of 3 bits, allowing for 7 combinations, that are compared on a simple equality test against the security classification code of the data. The assignment and maintenance of codes are the responsibility of either the Security Officer, the Data Administrator, or support personnel. The legal pairs of user codes and data classification codes are maintained in most cases in a system table which can only be accessed in executive mode. The table is core-resident except when it is part of a larger user profile and the number of users exceeds some convenient divisor of page size.

S2. USER ACCESS PRIVILEGES

If it is necessary to link individual users to subsets of the available data or processes, then some type of user profile must be developed either pointing to or specifically identifying the user access privileges. This profile may contain the identify or classification of the files available to the user, the manner in which the files can be accessed (read, write, process, modify or erase) the degree to which access is permitted, the specific terminals from which the user may operate, and the particular processes he may execute (named routines, standard jobs, or precompiled transactions). The creation and assignment of user profiles is the responsibility of some designated control group, which utilizes (in nearly every case) separate commands and support routines. The user profiles themselves are maintained as a system file, (although in a few systems, this was not the case) most probably resident on secondary storage because of its size. Because of its sensitivity, the data contents of the file are transformed in a few instances.

S3. PASSWORD

The password is the privileged identifier that a user must submit to obtain entry to the system. From a software standpoint, it is the only means of initially identifying a legal system user. Passwords may be required at LOG-IN, at both LOG-IN and LOG-OUT, or for every transaction executed. The more frequently the password is required, the less likely is the possibility that an illegal user will obtain entry but the more costly the user interface becomes in terms of its effect on the thruput and response time. Typed-in passwords range from 4 to 18 alphanumeric characters in

TABLE 5. C-1. SOFTWARE TECHNIQUES BY SYSTEM FUNCTION

File Handler
S24 - File Classification Code
S25 - File Access Lists by File
S26 - Deleted
S27 - File Access Lists by Level
S28 - File Access Profile
S29 - Data Element Classification Code
S30 - Data Element Profile
S31 - File Encryption - Single Key
S32 - * File Encryption - Multiple Keys
S33 - * Data Edition Number
S34 - Block-Write Collision
S35 - Ring Structures

Executive/Monitor
S12 - Privileged Instructions
S13 - * Relocatable Bootstrap
S14 - * Redundant Coding
S15 - * Module Dialogue
S16 - * Program Interpretation
S17 - Centralized IO Control
S18 - Error Monitors
S19 - Error Interrupts
S20 - * Exec Commands by Access Right
S21 - Boundary Maps
S22 - Memory Access Keys
S23 - Security Monitor

Terminal Subsystem
S7 - Error Correction Methods
S8 - Terminal Answerback
S9 - Terminal Profile
S10 - Terminal Character Suppression
S11 - Automatic Alarm and Disconnect

User Interface
S1 - User Security Clearance
S2 - User Access Privileges
S3 - Password
S4 - Password Dialogue
S5 - Consecutive Password List
S6 - Password Transforms

Others
S36 - Document Log
S37 - Limit No. of Erroneous Attempts
S38 - * Aggregate Techniques for Reports
S39 - Overwrite and Memory Erase
S40 - Classified Programs
S41 - Classification Header/Trailers on Hard Copy and Displays
S42 - File Log

* Not Now in Use

existing systems, are either fixed or variable in length and may contain blanks. No data was available on the format of voiceprint or key-pattern passwords. Passwords are not changed at all in some systems, are changed periodically in others, and are changed at irregular intervals in one proposed design. The more frequently the password is changed, the higher are both the maintenance cost and the error rate. Passwords are usually either assigned, generated, or selected using some standard random number generator, though non-standard variants were used in a few systems. As the number of combinations of characters in the password increases, and assuming that the number of users is constant, an increasing degree of assurance that legal passwords cannot be casually duplicated is obtained. Although it has been shown that any password scheme can eventually be broken, the degree of difficulty of doing so exceeds that of opening a 3-way 50-number safe-combination when the password exceeds 5 characters. Some systems attempt to detect password tinkering by assuming that a fixed number of consecutive illegal attempts (usually 2) from the same terminal is sufficient cause (see technique S37). Legal passwords are maintained in every case as a system table. Since the password in many systems is used as the key to locate user profile tables, it is coded in these instances in such a way as to be amenable to hashing algorithms. However, the range of available codes is then limited to reasonably uniform distributions of numbers.

S4. PASSWORD DIALOGUE

Since it is possible to eventually break any password scheme, several variations of the technique have been suggested to obtain more foolproof identification of legal users. One such variant is to require the user to engage in a form of dialogue with the system after the initial password is validated. This dialogue requires the user to provide responses either unique to himself (his payroll number in one case; another password in another case; a user defined item of personal knowledge in a third case) or to perform some relatively simple algorithm on either a system-supplied random variable or some transitory quantity (time of day, date, etc.); the system performs the same algorithm and checks the validity of the response. Once again, the scheme is susceptible to penetration, but the level of difficulty has been raised significantly — at a cost in increased terminal response time and communication line loading.

S5. CONSECUTIVE PASSWORD LIST

Another variation of the basic password technique is to assign a list of legal passwords to each user. The system will accept only the next password on the list each time that the user enters his password. This makes it extremely difficult to obtain a legal password through either passive deliberate penetration attempts or active tinkering with password combinations, but it also requires hard-copy lists of legal passwords to be made available to users and inevitably produces a greater number of erroneous entries by legal users. In spite of these drawbacks, consecutive passwords for each input has been accepted as an alternative to encrypted data links in one military system. A list of 64 passwords was chosen in all of the systems that implemented this technique.

S6. PASSWORD TRANSFORMS

Since the list of legal passwords is considered to be extremely sensitive information, it is usually resident in core, and is frequently also appended to program status blocks. Several systems have taken steps to prevent its being obtained either deliberately or accidentally by a readout from core. These steps involve implementing

various transformation techniques on passwords received. Huffman encoding is used in one system; a simple transposition of digits is used in another; an algorithm to produce non-reversible inversions is implemented in a third.

b. Terminal Subsystem

Nearly all of the systems surveyed provided for on-line terminals and a significant part of the software in these systems is that associated with terminal characteristics. Less significant, however, are the software techniques implemented to account for security requirements arising from remote terminals. The existence of on-line terminals, particularly at remote sites, introduces several potential security problems into the system. Terminals must be discretely identified to ensure that data is transmitted to the correct location. Either the users or the classes of users who can operate a given terminal have to be specified. Terminals at remote sites are susceptible to communication errors on transmission due to noise, may be easily expropriated for illegal use, and are subject frequently to public or semi-private display and the subsequent casual eavesdropping.

S7. ERROR CORRECTION METHODS

The most obvious problem with terminals connected to a system through communication lines is the noise factor on the communication lines themselves. This is more or less acute depending on the extent to which the terminal network is distributed and switched. It introduces the possibility that illegal values or erroneous addresses may be input in otherwise valid messages. Methods have been developed in many systems to reduce the effects of noise in transmission. They include hash totals (that is, cumulative adds) of characters or bits in the message, the totals either input with the message or determined by the controller and checked against the received totals; parity check bits and longitudinal redundancy checks (a parity word developed by a logical add of each word in the message) to reassemble garbled words; and retransmission to compare duplicate results, either simultaneously over duplex lines or serially over half-duplex lines.

S8. TERMINAL ANSWERBACK

Since it is possible to piggyback illegal terminals onto legal circuits, particularly in dialed and switched network systems, methods have been developed to uniquely identify legal terminals. In some cases, the identity is established by comparing the expected terminal address to a hard-wired terminal identifier that automatically transmits (i. e., "answers back") an identification-key (20 alphanumeric characters in the one system where this figure was known) with each input message or in response to a request code preceeding each output message. In other cases, the terminal user inputs a terminal identifier with his user password. In one system where the terminal location and function determine the level of data permissible, these terminal identifiers are changed each time that the system is brought up. The list of terminal identifiers is maintained as a privileged executive table in core, but not transformed in any system. Alternate terminal addresses are also frequently maintained, particularly for routing outputs; most systems will describe the system control terminal as an alternate address.

S9. TERMINAL PROFILE

The classes of data and/or users that can be legally associated with a given terminal are defined in a terminal profile list. This list is an extension of the terminal address table maintained by the executive. It usually only describes the highest security classification of data that can be output to a given terminal (using a 3-bit code in all systems where it was implemented). It may also include a list of specific transactions that can be executed from that terminal and/or a list of explicitly named users who may access through the terminal.

S10. TERMINAL CHARACTER SUPPRESSION

Any on-line terminal, remote or local, display or printing, that is used to input identification codes is susceptible to both casual and deliberate eavesdropping. The most obvious example of the latter is the implanting of bugs to read contents of the terminal buffer. Little can be done from a software standpoint to protect against this. However, there are two variations of a technique to reduce the vulnerability of input codes. In one case where hard-copy is used, the system strikes over the number of positions required for identifier codes each time such a code is expected from the user. This provides a marginal degree of protection. In other cases, where the hardware has the capability to receive it, a code is transmitted to the terminal to suspend printing or display-images on the text-line in which an identifier code is expected by the system. However, the characters must still appear in the terminal buffer area.

S11. AUTOMATIC ALARM AND DISCONNECT

If the system is able to detect that a terminal or a terminal connection is being used for (attempted) illegal input, it is necessary to alert the control group and to isolate the suspect terminal from the system. The alarm can be either an audible alarm, a light signal, or a printed message routed to the system control terminal. Obviously, the alert should include the terminal address (it doesn't in some of the systems reviewed). It could also include the nature of the attempted input. Since it may be advantageous not to alert the interloper that his presence has been detected, the isolation of the suspect terminal in one suggested plan would still permit it to remain linked to the system by engaging the user in a series of questions and delays. In most cases, however, the terminal is disconnected and/or the keyboard is locked to prevent further communication. Bringing the terminal on-line again usually requires that the security officer or control group input a special identifier code.

c. Executive/Monitor

The heart of any multiprogram system is the executive control routine. This routine accepts input requests, initiates jobs, assigns resources, maintains status tables, responds to interrupts and error conditions, requests data transfers, and monitors the activity at every system level. It is the most complex, sophisticated, and important component of the software. By its very nature, it is perhaps the most difficult to penetrate but then, again by its nature, it is undoubtedly the most rewarding. In this area in particular, procedural techniques must be relied upon. It is impossible to prevent support personnel from leaving "trapdoors" or potential entry points in the software and the need for the integrity of and confidence in support personnel is paramount. Many of the techniques developed in this area also require parallel hardware features.

S12. PRIVILEGED INSTRUCTIONS

Given that the hardware has a master/user mode capability, the set of instructions that can be executed in the master mode are regarded as privileged instructions. Since they are intimately involved with system control, e.g., the setting and resetting of bounds registers, the initiation of channel commands, the loading of read/write address registers, the diciphering of internal and external interrupts; they have an immediate application to security requirements. They should be used sparingly and should be concentrated in a few easily associated routines. The routines should operate in privileged mode as briefly as possible, branching to user mode to perform the function initiated. Dispersing privileged instructions in many executive routines simply improves the chances for trapdoors and illegal circumvention.

S13. RELOCATABLE BOOTSTRAP

If it were possible to bypass protection keys and to gain access to areas of memory normally reserved for the executive and its tables (perhaps through a trap door in another low-level executive function), then it would also be possible to read any of the access list and authority tables controlled by the executive. One technique suggested to reduce the likelihood of this occurring is to perform bootstrap loading of executive routines from a changing key address. In this manner, executive routines and tables no longer have absolute locations relative to each other and to the user partitions, and only haphazard location of the secured routines would be possible. This technique requires that executive routines be coded so that they can be link-edited, and that executive overlays be called in as relocatable code, requiring an absoluting step before execution. Because of its potential effect on the efficiency of the system, the technique has only been discussed (although one system did develop a prototype version of its executive in this fashion).

S14. REDUNDANT CODING

Since it is possible to modify code prior to loading (and, in a few cases, after loading), it has been suggested that key routines exist as multiple, discrete copies, and that requests for the services of these routines be executed in parallel by each copy. The results can then be compared, including number of instructions executed. The effective cost of this approach is high even in a multiprocessing system, but it further insures that key security routines cannot be modified or executed without detection.

S15. MODULE DIALOGUE

In any calling sequence, the parameters passed between modules are usually specified as a part of the standard call macro. It has been suggested that this be modified somewhat in those cases where it is feared that an interloper may substitute his own code for a system routine. At random points in the routine in question, private call parameters known only to the programmer responsible for that routine can be inserted. The routine called (or calling) is also prepared to expect the interspersed dialogue words. Since these would be difficult to detect in absolute code, it would raise considerably the level of difficulty associated with making such code substitutions.

S16. PROGRAM INTERPRETATION

Since it is difficult to detect subtle changes in absolute code, it has been suggested that programs be loaded through an interpreter at all times. If the interpreter includes some kind of code optimizer, each version of a program in its absoluted form would be slightly different than the preceding one. In this way, not only would it be difficult for a penetrator to modify or decipher program routines (except as a one-time event), but it would also be difficult for the programmer himself to take advantage of fixed relationships in his program that might permit the introduction of trap-doors.

S17. CENTRALIZED IO CONTROL

This is a fairly common technique employed by most third-generation systems. It separates application programs from direct address references to IO devices and instead requires them to submit macro commands that deal with the device as a logical, virtual, or relative extension of memory. The executive then generates and performs the physical IO commands and thereby is able to maintain control over boundary establishment and limited units of allocation. Without the equivalent hardware capability to trap to monitor mode whenever privileged instructions are attempted, this approach cannot be validated.

S18. ERROR MONITORS

In a system with many users, the cost of maintaining security can increase significantly if the user error-rate is high. This technique is intended to maintain a rating of the capability of individual users to perform the procedures associated with inputting valid transactions. If their error-rate increases beyond a predetermined level, then their accessibility to the system is decreased. The cost of maintaining this scheme is quite high, however, since it requires some corresponding method to re-evaluate and to certify the user's capability.

S19. ERROR INTERRUPTS

Any attempt to perform an illegal operation, to address some location outside of assigned boundaries, to input erroneous data, etc., should be the cause of an error interrupt. The routines to handle such interrupts can attempt to correct the error and resubmit the request, can abort or suspend the user in question, can alarm control authorities, or can regard the error as acceptable, flag it, and continue processing. Once it is determined in monitor mode what the interrupt is, any further processing to deal with it should be performed in user mode to reduce the occasions for illegal execution of privileged instructions.

S20. EXECUTIVE COMMANDS BY ACCESS RIGHTS

Since most executive request macros deal with abnormal processing situations (more memory required; additional file space allocated; suspend processing, communicate with the operator, debug commands, etc.), it is feasible to regard the execution of these commands as restricted to a controlled group of service routines. This technique associates a category code with all executive command routines and restricts their direct use to only the subset of users cleared to the equivalent category of access.

In its most sophisticated implementation, this technique led to the development of rings or layers of processing linkages (see technique S35).

S21. BOUNDARY MAPS

Boundary maps are the legal units of allocation assigned to given users. When associated with absoluted tasks, they represent the direct input to base and limit registers that determine the domain in which a user can be active. Boundary maps in most cases are stored with the user in question and represent a potential means to illegally extend the the accessibility afforded a given function should they be modified.

S22. MEMORY ACCESS KEYS

In a page or segment-oriented system, there are usually lock registers associated with each physical page in memory. When a user is assigned to memory, is identifier is used to generate a unique key that is loaded into all of the page registers assigned to the particular user. An address reference to the protected pages cannot be made unless it contains the appropriate key-pattern in its own key register. Obviously, selected executive routines must have a universal key. Setting and access to the key registers should be a privileged function.

S23. SECURITY MONITOR

The security monitor is a technique that attempts to certify the validity of the various protection mechanisms in a system. At its simplest, it consists of a set of on-line diagnostic routines that exercise the various hardware components in a configuration, expecting a valid operation to produce a pre-designated result. At its most complex, it attempts to deliberately execute illegal hardware or software operations and then to determine whether or not the responsible protection mechanism has successfully intercepted and handled the illegal attempt. The more complex version can have a significant effect on system thruput and, therefore, requires a careful consideration of what are acceptable and expected failure levels.

d. File Handler

The data available in any system is the reward for penetrating the system. The data available in an automated system significantly increases the potential reward because of the large amount, the focal and functional concentration, the anonymity of access, and the difficulty of detection. Data protection is traditionally obtained by assigning responsibility and limiting access. Techniques to accomplish both are available in automated systems. At least the same level of protection can be obtained in an automated system as in a manual system.

S24. FILE CLASSIFICATION CODE

This technique is commonly employed in most systems that deal with formally classified data. It simply involves assigning one of the categories of classification to each data file (in this case, the code linked with the file description is usually a literal character - T, S, C, or U - and not a 3-bit configuration) and then either assigning the file only to jobs or individuals of equal or higher clearance, or, in the case of shared files,

releasing data from the file only to users of equal or higher clearance. A somewhat adventuresome extension of this technique is to attempt to automatically assign classification levels to new files. In one system, this is done by using the highest classification from contributing files. In a proposed scheme, it is done by doing a key-word count and weighting the file in accordance with the number of key-words encountered. In neither case was it shown to be statistically more or less effective than manual classification, except in the marginal area between unclassified and confidential.

S25. FILE ACCESS LISTS BY FILE

An extension of the file classification code is the assignment of specific authority lists to each file. These lists describe the original creator (or owner) of the file, other individuals, groups, terminals, etc., who can share the file, usually the manner in which they can access the file (read, write, modify, execute, or erase), and in a few cases, the degree to which access is permitted. At the file level only, it corresponds to the cataloguing function of most third-generation systems. The assignment and change of those lists is either available to the owner of the file (as in most of the operating system catalogues) or to the data administrator (as in most data management catalogues).

S26. (Deleted entry)

S27. FILE ACCESS LISTS BY LEVEL

An extension of the previous technique that permits it to be used in a more flexible environment than that described by formal discrete files is to assign access lists to levels of files, or to individual nodes in data sets. This is especially useful if the files consist of programs arranged in some kind of hierarchy from common (free) utility routines to machine-oriented (owned) system routines. The cost of maintenance, particularly the determination of who can access what at which level, is quite high. In one system maintenance problems led users to avoid the protection provided.

S28. FILE ACCESS PROFILE

If a number of users with different need-to-know interact with a shared set of data, it is necessary to distinguish the data rights of each user. This is accomplished by assigning to each file descriptor a profile word that contains a set of flag-bits, each flag-bit representing a unique need-to-know identifier. In the systems utilizing this technique, separate profile words are assigned for read and update access. There is associated with each user's profile an equivalent word with the need-to-know flag. assignments according to his requirement for data. A one-to-one correspondence between user and file profiles at each flag position is required before access is permitted.

S29. DATA ELEMENT CLASSIFICATION CODE

This is identical to the file classification code (S24) except that each data element in the file is separately classified. The system can then handle files with mixed classes of data. This feature greatly reduces the redundancy associated with file processing since it permits the grouping of data by functional purpose and utilization rather than by

classification. However, it raises both the cost of creating files and the cost of assigning and maintaining categories of classification.

S30. DATA ELEMENT PROFILE

This technique is identical to file access profile (S28) except that the system can now discriminate among need-to-know at a finer level of detail. It is usually implemented by assigning an update word and an access word to each data element descriptor, these words having particular bit settings according to their class. Users with matching need-to-know patterns are permitted access to the data element. The bit-patterns are combined to form a composite need-to-know profile for each data request. In a few instances, the data element profile also contains legal values of a data element that are accessible by a given class of users.

S31. FILE ENCRYPTION, SINGLE KEY

Techniques for encrypting data have been suggested for use in file handling systems. In most cases, these are variations of cryptographic techniques applied to communication transmission, and consist of applying a single key to all records in the file. However, because of the large number of records in most data files and because of the rather consistant pattern of field occurrences, this type of file encryption only provides a marginal increase in protection. Depending on the amount of character manipulation in the crypto technique CPU thruput costs can be quite high. In one system that offers this service as an option, most users choose not to avail themselves of it because of the overhead.

S32. FILE ENCRYPTION, MULTIPLE KEYS

A variation of the preceeding technique that reduces the possibility of deciphering is to use a different key (either a cascading or random number sequence) for each record or for certain numbers of records. This breaks the consistency of the encoded data and does not significantly affect the cost of the encoding/decoding process.

S33. DATA EDITION NUMBER

In a system where multiple users are concurrently updating a set of shared data files, it is necessary to prevent one update sequence from intruding on another. A suggested technique is to assign an edition number to every record in the data base. Each user contains the latest version of this edition number in its associated data buffer. When a retrieved record is to be written back into the data base, the file handler checks the user's edition-number against the data base edition number and only permits the update if the edition numbers are the same. The file handler also updates the edition number.

S34. BLOCK-WRITE COLLISION

This technique addresses itself to the same problem as above, but does not attempt to control the interaction at the record level. Instead, in the systems in which it is implemented it is keyed by a block-busy flag assigned to each file segment. When a segment is retrieved for update, the block-busy flag is set, as are all antecedent blocks

in the structure (or only the highest level block if the entry point is always top-down through the same index). The busy-flag is left on until the user has indicated completion of the update and the file-handler, if necessary, has modified the affected index blocks.

S35. RING STRUCTURES

Ring-structures are a combination of logical layers, or rings of data grouped by sensitivity, and identifiers associated with each user that describes the equivalent sensitivity of the user. It is permissable for a user to access and/or execute any data or routine in its own ring. When a call is made to a segment in another ring, the system traps to a gate controller, which determines if the called ring is more or less sensitive. If the sensitivity is less, then the call is linked to the ring in question. If it is greater, then a check is made of the access list associated with the requested segment. This list identifies legal users (or classes of users), and indicates the type of access and the particular entry point at which they may use the requested segment. The gate controller then establishes the required linkages. To prevent repetitive calls to the gate controller, upper and lower bounds can be assigned to each type of access for any user; requests to any rings within those bounds are automatic and equivalent to operating within the ring of the requesting user. The system that implemented this technique had special hardware registers to check the ring-brackets of segment requests.

e. Others

Various other software techniques were encountered during the data collection phase that do not conveniently fit into the preceeding categories. For the most part, these techniques have to do with using the automated system to simplify or extend some of the procedural requirements in a secured system.

S36. DOCUMENT LOG

In several of the systems reviewed, particularly status and intelligence repeating systems, the system automatically maintains accountability log making an entry each time that a classified report is released to a user. This log includes the date and time of the original request, the parameters specifying the report extraction criteria, and the terminal and user identification. The Security Log is available only to an identified security officer.

S37. LIMIT NUMBER OF ERRONEOUS ATTEMPTS

This technique is applied at several intersection points between a user request and system functions. Since a potential interloper can tinker with legality checks at any one of these points, it is necessary to set some limit on the number of consecutive illegal inputs that will be accepted from any user. Some type of on-line monitoring is required to record and link the sequence of requests.

S38. AGGREGATE TECHNIQUES FOR REPORTS

A serious problem in on-line systems is the possibility that even though a user cleared to a low level of access can only access data legally classified at or below his level, the aggregation of all data accessed can provide the basis for interpretive conclusions about higher classified information. Techniques have been suggested that would combine the access profiles (see technique S30) at all data elements contained in the report into a new profile that would yield a restricted classification of the report on a need-to-know basis. However, this cannot prevent inferences from data, and more work is needed in this area to determine if some kind of weighting of information content might be possible

S39. OVERWRITE AND MEMORY ERASE

Any magnetic recording medium retains an electromagnetic image of the recorded data for some time after the initial impression. This residue can be read directly, albeit inadvertently, if access to the area is obtained or picked up through passive deliberate penetration attempts. Since both primary and secondary storage in most on-line multi-user systems is considered to be virtual memory, it is entirely possible that an area in which classified data had been stored and processed could be reassigned to a user having a lower classification level. To prevent this, methods have been developed to overwrite primary memory by cascading or leapfrogging thru the area and writing a system constant (usually zeros) after the memory space is deallocated. The confidence in this technique is increased if it is procedurally established that every user routine fills its scratch area with a different constant. In only a few systems is the same approach used for secondary storage, since the time required to overwrite deallocated file space on a peripheral device, particularly one with a single read-write head, can be considerable. If a centralized data manager is used by all system users for handling data files, it is conceivable that reallocated space can safely be maintained as "dirty" storage because it is not logically valid to read empty file space.

S40. CLASSIFIED PROGRAMS

This technique is used in batch oriented systems where a user consists of a set of programs and their associated data files. Since the programs are designed to suit this single set of data, they take on the classification of the data and can only be called or modified by job control statements input with the proper classification leader.

S41. CLASSIFICATION HEADERS AND TRAILERS ON HARDCOPY AND DISPLAYS

This technique is an extension of the current procedural technique of stamping at the top and bottom of every classified page of a report the classification level. It is usually a parameter option in the report generation routine.

S42. FILE LOG

The file log is an extension of the Security Log in which every reference to classified data is logged. It can include the previous data image if the reference causes a change. It also usually includes the terminal, user, time, data, and data parameters that initiated to reference.

2. Hardware Techniques

Many of the hardware techniques required for security purposes have been implemented in third generation commercial computer systems and in some military computer systems for the purpose of process control and data protection in a multi-programming and/or multi-processing environment. Military computer systems now in the prototype or development stage have available multi-programming/multi-processing techniques that are applicable to security requirements. The following paragraphs identify and describe these techniques plus others that have been suggested but not implemented. Figure 5. C-2 lists the techniques categorized by the major computer subsystem.

a. Central Processing Unit (CPU)

Security related techniques in the CPU provide control of the logical processes to access and change data. Techniques that isolate and control the operation of programs and access to data, provide for recovery from hardware failure, and allow centralized error checking of programs and data access are also applicable to the security of the automated system.

H-1 PROCESSOR MODE, PRIVILEGED INSTRUCTION SET

Present third generation computer systems have implemented multiple modes of operations differing in the ability to process available instructions and in memory access restrictions. Typically, the system may operate in one of two types of modes — the control or executive mode or a user mode. The processor will not execute a privileged instruction unless a Processor Mode Register is set to the control mode. In the control mode, all instructions can be executed and all memory accessed. In the user mode, privileged instructions cannot be executed and memory accesses are restricted to those which were assigned while the processor was in the control mode. Should a privileged instruction occur in a user program or a memory access attempted outside the allocated area, an automatic interrupt is generated returning control to the executive program. User programmed entry into the control mode is possible only by an unmasked interrupt. Programmed exit from control mode back to user mode is accomplished by executing a return-to-user mode privileged instruction. Any programmed software attempt to circumvent the interrupt entry is prevented in normal operation by physical lockout of access paths and flip-flops. Typical privileged instructions include:

a. Input/output (commands and register loading)
b. Memory bound register loading
c. Interrupt mask register control
d. Mode control register resetting to user mode

Restricting all input/output command and register loading to the control mode insures that all access to data and programs is controlled by the executive program. Similarly the control of all interrupts and core memory accesses is controlled by restricting the ability to load memory bound register, interrupt mask register, and mode control register to the control mode.

H-2 CORE MEMORY BOUNDING (BAR, Lock and Key, Paging)

Three major hardware techniques are used to limit core memory access of user programs to a bounded or allocated area established by the executive program. The base

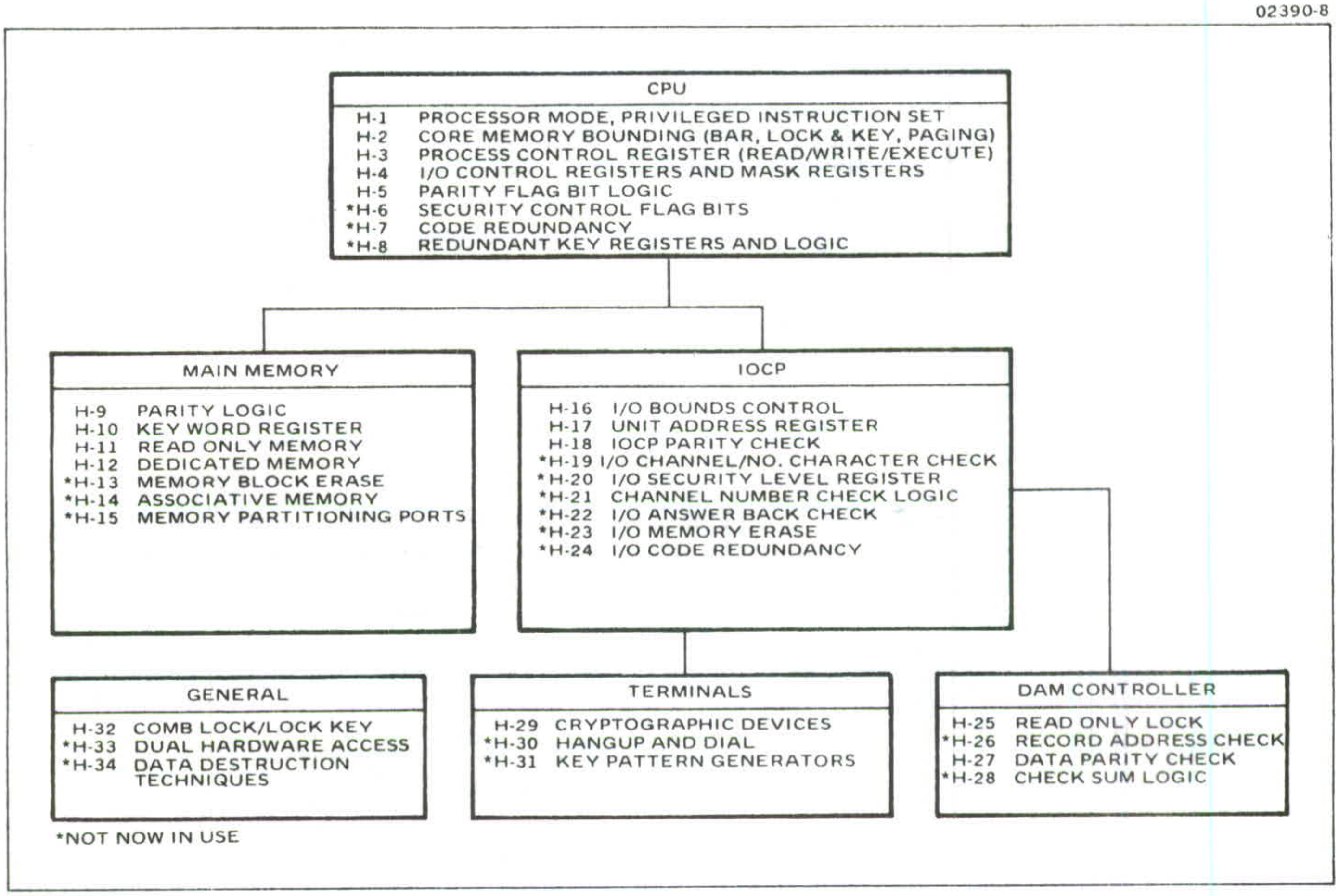

Figure 5.C-2. Hardware Techniques Applicable to Security

address register (BAR) containing upper and lower limits of allowable core memory access is used to insure that after indexing or indirect memory addressing, the hardware memory address is within the bounds of core memory assigned by the executive program. The lock and key memory bounds technique is implemented by the executive assignment of a key word to user programs and to memory areas that defines the authorized core access area for that user program. The key word is automatically checked in the core memory before any access is allowed. An unauthorized command or memory access request causes an interrupt to the executive. The third method, paging, uses the key word designated by the executive as part of the hardware memory address and access is physically impossible outside the bounded core memory area. The techniques vary in terms of hardware cost, the base address register the lowest; lock and key intermediate and paging the highest.

H-3 PROCESS CONTROL REGISTER (Read/Write/Execute)

Process control or the control of the right to read or write data or to execute data has been implemented in accordance with the basic method used for memory bounding. Flag bits are used in the associated memory bounds register to indicate the rights of the user program to read and/or write into core and to execute program instructions in a given memory area. Any attempt to read/write/execute by a user program is automatically checked against the memory bounds control word and unauthorized attempts cause an interrupt to the control mode.

H-4 I/O CONTROL REGISTERS AND MASK REGISTER

The loading of I/O control registers and I/O mask register allows the centralization of all input/output to executive control which is essential to the effective isolation of user programs and data. The mask register provides an effective means of controlling different types of interrupts including those associated with inadvertent or deliberate attempts of user programs to perform unauthorized actions.

H-5 PARITY FLAG BIT LOGIC

Parity generated for the transmission of data and is checked by receiving units. A single parity bit detects any single or odd number of bit errors in the word (character or bit group) in which it is included. It is used in most third generation computers to provide a method to detect hardware errors in all parts of the computer system. Detection of a parity error causes an interrupt to the executive mode.

H-6 SECURITY CONTROL FLAG BITS

Use of flag bits in programs and data for the purpose of indicating security level is a technique which has been suggested but not implemented in any of the systems surveyed. The flag bits in each data word would indicate the security level of the data and the class of user who could read or write the data word. The flag bits in program instructions would govern execution of the program. Both hardware and/or software control has been suggested to interpret the flag bits. The high cost in additional memory to store these extra bits has been cited as one of the reasons why the technique has not been implemented. Some of the more advanced systems have implemented the use of flag bits with software interpretation at the control word level rather than the individual word level.

H-7 CODE REDUNDANCY

The use of extra bits to provide code redundancy to enhance the capability to error correct or to better identify errors has been suggested but not implemented for key control instructions. The mode control register and memory bounds register have been suggested as areas where code redundancy should be used.

H-8 REDUNDANT KEY REGISTERS AND LOGIC

The technique of using redundant registers for mode control and I/O channel control has been suggested but not implemented. The use of multiple registers that would cause an error interrupt if they did not agree would insure the proper functioning of these key controls.

b. Main Memory Module

The control of access to the main memory module is necessary for proper security control in an automated system. All programs and data that will be accessed must ultimately reside in the main memory. Control of main memory areas in some systems is accomplished by the CPU, in others, by a combination of CPU and memory circuits. Since all data and programs are read from the memory, the integrity of data in the memory is paramount. Hardware failures must be detected to prevent the possibility of CPU execution of changed code with unpredictable results.

H-9 PARITY LOGIC

The memory unit checks and/or generates parity bits for storage in order that odd bit failures can be detected. A parity failure causes an interrupt to the executive mode in the CPU.

H-10 KEY WORD REGISTER

In lock and key memory protect systems, the memory compares the key flags of the access request with those set by the executive in the Key Word Register. This prevents unauthorized read/write/execute access to memory for data or programs.

H-11 READ ONLY MEMORY

Read only memory is used in some systems for key control programs to provide protection against unauthorized change to programs or data. The higher cost of such memories has limited their use.

H-12 DEDICATED MEMORY

The use of separate memories for different classes of users has been suggested to provide security of classified data. Such a system requires control by an executive program. The physical separation of data into separate dedicated memories also allows correlation software techniques to code data. Plug-in dedicated memories for special programs and data such as security access lists and security monitor programs, have been proposed but not implemented.

H-13 MEMORY BLOCK ERASE

A special instruction and associated hardware to clear a specified block of memory has been suggested in order to clear residue from a task upon completion of that task. This procedure would insure that classified data is destroyed before memory is reallocated. This prevents core dump instructions at the beginning of the new user program from outputting the previous user's data.

H-14 ASSOCIATIVE MEMORY

An associative memory allows the retrieval of data or programs based upon a code match rather than a hardware address. It has been suggested as a technique that could be used to retrieve data or programs based upon a code designation of the data or program class. Associative memories have not been implemented in systems because of their high cost.

H-15 MEMORY PARTITIONING PORTS

The use of special dedicated ports or paths into dedicated blocks of memory has been suggested as a method to isolated special classes of data. The special ports could only be accessed by a special set of privileged instructions.

c. Input/Output Control Processor

The input/output control processor (IOCP) provides the hardware interface with mass memory (disc, drums, tapes, card reader/punch) and with the system user (printer, display, communications lines, etc.). Data is transferred between those external units through the IOCP to the CPU and main memory. The IOCP features which are particularly important to data security are the registers and logic that route the data to/from the proper external device from/to the proper main memory core block and CPU.

H-16 I/O BOUNDS CONTROL

The CPU provides the starting address and either the word count or ending address for any data transfer to/from external devices and main memory. The IOCP, through bounds control, insures that the data is transferred to/from the allocated block of memory. Each address is automatically checked to insure that it is within the address bounds. Any address outside the address block causes an error interrupt to the executive mode.

H-17 UNIT ADDRESS REGISTER

The unit address for outputs of data is furnished to the IOCP by the CPU. Unit select gates connect the appropriate unit buffer in the IOCP based upon the contents of the Unit Address Register. For data input, demand queueing is processed by the CPU which furnishes the IOCP with a control word defining the allocated memory block for the input.

H-18 IOCP PARITY CHECK

Interequipment address and data transfers and unit address control word parity checks provide the capability of detecting single failures and prevent misrouting of data.

H-19 I/O CHANNEL/NUMBER CHARACTER CHECK

Logic has been suggested but not implemented to provide a means to identify misrouting of data in the IOCP. Before any data is released to a channel, the channel number terminal device would be checked by the CPU. A character count register would be set to allow transmission of a specified number of characters and decremented to zero as characters are transmitted. At zero count, the channel number would again be checked and the character count register reset. This procedure insures that data is being input/output on the correct channel and, in case of malfunction, limits the amount of data released.

H-20 I/O SECURITY LEVEL REGISTER

This I/O register would check a record control security code word against a channel security level code. If the classification level of the control word was higher than the channel level, an interrupt would be generated. This channel security level check has been suggested but not implemented.

H-21 CHANNEL NUMBER CHECK LOGIC

This technique requires that the channel control word from the CPU to the IOCP be transmitted twice and matched by dual registers in the IOCP before data is transmitted on the channel. A mismatch causes an error interrupt. The technique has been suggested but not implemented.

H-22 I/O ANSWER BACK CHECK

This suggested technique utilizes hardware which requires answer back identifying the receiving terminal unit before allowing any transmission to the terminal unit. The answer back terminal unit identification is checked for match against the original control word in the IOCP before data is transmitted.

H-23 I/O MEMORY ERASER

The IOCP provides the capability to clear a block of memory of residue from a previous use of the space. A control word from the CPU specifies the main memory block address. The IOCP then cycles through the block addresses transmitting all zeros to the lock addresses. At the end of the block, the IOCP generates an interrupt to the CPU that identifies the block as being cleared. This technique could be used in CPU limited systems instead of a software routine.

H-24 I/O CODE REDUNDANCY

Additional bits over those logically required could be used for terminal addressing. This would provide the capability of error detection and error correction. Additional hardware error detection and error correction hardware would be required.

d. Direct Access Memory (DAM) Controller

The majority of the data and programs is stored on direct access memory (disc, drum, tape) and transferred to the main memory when required for processing. Physical and electrical control of access to these devices is necessary to insure security. Electrical access to the devices is through their controllers which provide for record address location and read/write execution on the device.

H-25 READ ONLY LOCK

Logic and switches on the controller provide the capability to allow only read on specified tracks of disc and drum. The switches are set to read only or read/write at system setup time. Tape drivers can also be set to read only by write disable switches.

H-26 RECORD ADDRESS CHECK

The controller requires a confirmation check of the record address from the IOCP before read/write is enabled.

H-27 DATA PARITY CHECK

The controller checks for parity of each word as it is read from the device. A parity error generates an error interrupt to the IOCP.

H-28 CHECK SUM LOGIC

The controller counts the bits in a given record and checks this total against a total entered at the beginning or end of record. If the total does not agree, an error interrupt is generated. This technique allows the detection of unauthorized change to records.

e. Remote and Local User Terminals

The computer system terminals are the means used for communication between the automated system and the user. The access to and user capabilities allowed at the terminals are key security control features. The hardware security techniques identified at the terminals provide means to limit access and control user capabilities.

H-29 CRYPTOGRAPHIC DEVICES

Cryptographic devices are used to automatically encode and decode data on communications channels. Such encoding provides protection against electromagnetic radiation of communications channels and wiretapping. The techniques used in the devices are

highly classified and require a special engineering discipline. This survey did not attempt to identify individual techniques and their value, as numerous other classified studies have been devoted to this area. The survey was limited to the identification of systems where such devices were used.

Such devices could be used not only for communication channel security but also to encode data within the computer system. However, the computer system using software techniques can implement any of the hardware techniques. Therefore, it becomes a cost-effective decision as to the use of cryptographic hardware devices. For communication channels, cryptographic techniques are the only known practical method to prevent access to data by radiation or wire tapping.

H-30 HANG UP AND DIAL

This technique provides logic that transmits a request from the terminal for computer services. The computer then requests a verification of the request from the terminal. An identification code is automatically sent that confirms the terminal request. Separate lines have been suggested for the transmission of the two identification requests. In some systems, a telephone confirmation by computer support personnel is used for the verification of the on-line terminal request.

H-31 KEY PATTERN GENERATORS

Several techniques to identify individual users have been suggested. Identification card readers are used by a few systems. Voice, fingerprint, and combination lock code generators have been suggested but not implemented for the generation of individual key patterns. The key patterns are transmitted to the computer system where access rights to data and programs are authorized on the basis of the key pattern comparison.

f. General Techniques

Some of the hardware techniques apply to more than one of the subsystems of the computer. They are described in this section.

H-32 COMBINATION LOCK OR LOCK AND KEY

The physical securing of key parts of the computer system by combination lock or lock and key has been suggested as a method to limit access to critical circuits. The circuits suggested for this protection are the power circuits at terminals, read only switches on mass memory devices, the cabinets that contain the IOCP, CPU, core memories, and dedicated plug in memories.

H-33 DUAL HARDWARE ACCESS

This technique would require the simultaneous insertion of keys by more than one person to gain access to key hardware components.

H-34 DATA DESTRUCTION TECHNIQUES

The problem of destroying classified data in a short time in cases of seizure of a computer area or a remote terminal has been discussed, however, no techniques were suggested in the literature for application to this area of security control. Seizure is a security problem in both automated and manual systems. The problem is not an easy one since large volumes of data must be destroyed in a short time. The sequential writing of a random number stream on to data files does not prevent residual effects on magnetic storage devices, but would make data recovery much more difficult. Physical destruction of devices, depending upon the level of destruction, could destroy the data involved. The degausing of the mass storage devices is a possibility but could require too much time or unreasonable power levels. It is felt that further study is needed to identify reasonable and practical methods to provide for protection from the threat of area seizure.

3. Procedural Techniques

Procedural techniques are required to set up, maintain and monitor the automated security system. They apply as well to protecting data in the form of hard copy reports as they do to protecting it in the form of backup tapes and disk packs, program listings, program decks, common data pools, and user ID-lists and passwords. They are needed to establish the manual as well as the automated methods by which the four functions of security are accomplished. These four functions are classifying and declassifying data, providing the means to safeguard the data, providing for proper accountability, and allowing the dissemination of the data on the basis of a need-to-know.

The security that is obtained in any system utlimately rests on the responsibility and trustworthiness of the indivudals who are associated with it. There are, therefore, two primary procedural techniques that transcend the four functional area and apply equally to every aspect of system activity. The first is to guarantee that the authority to set up, maintain and monitor the system is accomplished only by those designated to perform the indicated functions. The second is the formal establishment by law of personnel responsibility for the safeguarding and dissemination of classified data. Each person is responsible to safeguard classified data or programs made available to him for the performance of his official duties and to limit dissemination of that data to only those with proper security clearance and need-to-know. These two principle procedures are implemented formally in all government and in several commercial systems reviewed.

Additional major procedural techniques that are applicable to one or another of the four security functions are outlined in Table 5.C-3 and explained in the following paragraphs. The same numbering scheme is used to identify procedural techniques as was used for hardware and software. However, because the procedural techniques are self-evident, they are not described individually, but are discussed as a class.

a. Classifying and declassifying procedures. Procedures are available that allow the assignment of security classification to designated individuals at the file and program level. In some systems the User Profile Table specifically allows the authority to classify/declassify given files. In all systems, the ability to classify/declassify files and programs is only permitted to the individual of highest level security authority in the installation usually the system security officer.

b. Safeguarding Procedures. All systems use secure area protection for central computer areas and remote classified terminals. Access lists are maintained to

allow entry into these areas. The establishment of such access lists is the designated responsibility of the system security officer through formal submittal to the security authority. In addition, this officer is charged with the establishment and maintenance of user/terminal profile tables that provide the authority to access data and files and the assignment of personal ID and/or code words to authorized users. These should preferrably be assigned in a random manner and at nonperiodic intervals. Other procedural technique used are the partitioning of knowledge and responsibility for file and program maintenance, the use of software debug procedures that required several levels of check out and review before the program is accepted for system use, the forbidding of program changes to the author-programmer, the centralization of protection mechanism and reduction and control of the use of the privileged mode for system operation, and finally higher level concurrence and approval for system related changes to software or hardware.

c. Accountability. Procedures are available to provide periodic review of access logs, security monitor logs, record counts/check totals and file logs. Such reviews are the responsibility of the system security officer or data administrator and are called for at stated periodic times. Document signout procedures similar to that used in manual systems for classified data are used for hard copy classified material. An inventory of all hard-copy classified material back-up tapes disk, packs, program listings and card-decks is conducted on a periodic basis.

d. Dissemination. The dissemination of data in automated systems is based on user/terminal profile tables in some systems and in others on the use of access lists. The system security officer is responsible for the preparation and maintenance of the tables and lists. In systems that use passwords or code words, procedures are established for the dissemination of the current codes on a periodic basis to those authorized users.

These procedural techniques are applicable to any system, since they are a common requirement for providing adequate protection. Consequently, all of the techniques shown in Table 5. C-3 are applicable to the TACC, regardless of the final design decisions made.

D. CATEGORIZING TECHNIQUES BY SECURITY THREAT

The following paragraphs categorize security techniques by their applicability for countering classes of security threats. Table 5. D-1 lists the security threats and security techniques or countermesaures to defend against each penetration threat. The techniques are identified by the key-numbers assigned in Section 5. C.

1. File and Program Access.

a. Browsing. Techniques to prevent browsing (the attempt by an authorized user to obtain unauthorized access to data or programs) limit user access to main memory and files. Applicable hardware techniques centralize main memory allocation, centralize input/output, and limit access to programs and data to allocated memory blocks. Software techniques determine access rights and allow access to programs and files only on the basis of access rights. Both hardware and software techniques provide positive identification of the user. Both provide protection against the access to programs or data of a higher security classification than the security clearance of the user.

TABLE 5. C-3. PROCEDURAL TECHNIQUES CATEGORIZED BY SECURITY FUNCTION

Security Function	Security Procedure
Classification and Declassification	P1. Higher level concurrence for change.
	P2. File and program level assignment.
	P3. User profile right to assign and change security level.
Safeguarding	P1. Higher level concurrence for change.
	P4. Secure area.
	P5. User/terminal profile table.
	P6. Partitioning of knowledge and responsibility.
	P7. Standardized debug discipline.
	P8. Fragmentation of file and software responsibility.
	P9. Limited access to key areas
	P10. Localization of protection mechanism.
	P11. Reduction and control of the use of priviledged mode.
	P12. Establishment and maintenance of access lists.
	P13. Password and access code assignment.
Accountability	P14. File/Data element access log review.
	P15. Security monitor log review.
	P16. Record counts, check total, file log reviews.
	P17. Document sign-out.
	P18. Periodic inventory.
Dissemination	P5. User/Terminal profile tables.
	P12. Establishment and maintenance of access lists.
	P13. Password and access code assignment.
All of the above	P19. Designated Authority.
	P20. Personnel responsibility.

b. Masquerading. Techniques to prevent masquerading (an unauthorized user attempting to act as an authorized user) are in the class that provide positive identification of the user. Hardware techniques such as key pattern generators and answer-back checks are in the class. Software techniques include password and user access privilege tables.

c. Piggy Back Entry. Penetration by connecting to communications lines and using the right of an authorized user after he has identified himself to the system can only be prevented by communications encyption or by securing the communications lines and securing areas where wire tapping could be installed.

d. Copying. Unauthorized copying of data can be prevented by isolating memory access, centralizing processing control based upon user identification and user access privileges, erasing main memory core to prevent read out of residue, and by coding of data so that it cannot be decoded unless the user has the right to access decoding programs.

e. Changing. Unauthorized change to data or programs can be prevented by techniques that limit write privileges. They depend upon proper identification of the user and user write privilege determination from user profile tables.

f. Theft. Techniques to prevent theft consist of lock and key securing of memory devices and coding of data to prevent read out if theft is successful.

g. User Error. Inadvertent user error can cause unauthorized release of data. Techniques that centralize process control and memory access prevent such release of data.

h. Hardware Failure. The use of parity check circuits, redundant key circuits, software error monitors and on-line diagnostics provide protection from hardware error.

i. Software Failure. Centralized process control plus memory partitioning and software error monitors provide protection from software errors.

j. Trap Door Entry. The creation of trap doors by support personnel that allow circumvention of security techniques is difficult to prevent. Procedural techniques are the most effective against this threat. The centralization of process control, memory partitioning, and limiting authority to change programs provides some protection. The use of advanced software techniques that isolate programs and their operation from each other are expensive in thruput, (Refer to Section 5.C; S13, S14, S15, S16).

k. Support Entry. Procedural techniques are the most effective against this threat. Automated techniques that allow isolation of access areas provide some protection.

l. Area Seizure. Coding of data, data destruction, and degaussing techniques are possible. This area should be investigated further to establish realistic techniques. Procedural techniques are effective in preventing area seizure.

2. Hardware Access.

The ability to deactivate protection circuits, change data routing, and install bugs can only be controlled by securing and protecting the area involved. Hardware lock and key and dual access can be used to provide protection.

TABLE 5.D-1. AUTOMATED SECURITY TECHNIQUES APPLICABLE TO SECURITY THREATS

Security Threats		Automated Security Techniques		Remarks
		Hardware**	Software*	
FILES AND PROGRAMS (TERMINAL ENTRY)	Browsing	H-1, H-2, H-3, H-4, H-6, H-10, H-12, H-14, H-15, H-16, H-20, H-29, H-31	S1, S2, S9, S12, S17, S21, S22, S23, S24, S25, S27, S28, S29, S30, S31 S32, S37, S40, S59	Access rights, user identification, security protection techniques needed.
	Masquerading	H-17, H-22, H-30, H-31 H-32, H-33	S2, S3, S4, S8, S10, S11, S23, S37	User identification techniques needed.
	Piggy Back Entry	H-29, H-32		Secure Comm lines, secure areas.
	Copying	H-1, H-2, H-3, H-12, H-13, H-16, H-23, H-29	S1, S2, S12, S24, S25, S27, S28, S29, S30, S31, S32, S33, S39	Limit access by user I. D., centralize process and memory control.
	Changing	H-3, H-11, H-25, H-28, H-29, H-32, H-33	S2, S7, S9, S25, S26, S27, S28, S31, S32, S33, S34, S35	Implement read only techniques.
	Theft	H-29	S31, S32	Code all data, secure area, use lock and key.
	User Error	H-1, H-2, H-3, H-4, H-16, H-29	S2, S11, S12, S18, S19, S20, S23, S24, S25, S27, S28, S29, S30, S31, S32, S37, S39, S36, S42	Centralize process and memory control.
	Hardware Failure	H-1, H-2, H-3, H-5, H-7, H-8, H-9, H18, H-26, H-27, H-29	S9, S12, S17, S18, S19, S23	Implement fail safe hardware and software.

*Refer to Section 5. C. 1
**Refer to Section 5. C. 2

TABLE 5.D-1. AUTOMATED SECURITY TECHNIQUES APPLICABLE TO SECURITY THREATS (Sheet 2 of 2)

Security Threats		Automated Security Techniques: Hardware**	Automated Security Techniques: Software*	Remarks
FILES AND PROGRAMS (TERMINAL ENTRY) cont'd.	Software Failure	H-1, H-2, H-3, H-4, H-16 H-23, H-29	S9, S12, S17, S18, S19, S23, S36, S42	Centralize process and memory control. implement fail safe software
	Trap Door Entry	H-1, H-2, H-3, H-4, H-6, H-17	S12, S13, S14, S15, S16, S22	Centralized process and memory control limits threat, Trustworthy support personnel with area security needed.
	"Support" Entry	H-1, H-2, H-3, H-4, H-6, H-10, H-12, H-13, H-14 H-15, H-20, H-23, H-29, H-31, H-32, H-33	S2, S3, S4, S5, S6, S9, S12, S13, S14, S15, S16, S17, S21, S22, S23, S25 S27, S28, S36, S37, S38, S39, S40, S42	Centralized process and memory control limits threat. Partition access to programs and equipment.
	Area Seizure	H-29, H-34	S31, S32, S39	Needs more investigation to identify practical techniques.
HARDWARE CPU IOCP MEMORY TERMINALS	Deactivation of Protect Circuits	H-32, H-33		Trustworthy support personnel, Secure area, lock and key, and dual access limits threat.
	Changing Data Routing	H-32, H-33		
	Installation of Bugs	H-32, H-33		
COMMUNICATIONS	Electromagnetic Radiation	H-29	S31, S32	Cryptographic lock and key and secure area only known techniques.
	Wire Tapping	H-29, H-32	S31, S32	
	Install Bugs	H-29, H-32		

*Refer to Section 5. C. 1
**Refer to Section 5. C. 2

3. Communications Lines Access

Cryptographic techniques plus area protection is the only known method to prevent wire tapping, the monitoring of electromagnetic radiation, and the installation of bugs.

E. CATEGORIZING TECHNIQUES BY SECURITY REQUIREMENTS

In this section the techniques discribed in previous sections are associated with system design features that reflect the security requirements of the system. Applicable techniques that provide for proper security are listed in Table 5. E-1. For each class of user language capability, terminal location and use, and user/data clearance level. Since each system has three design features related to security requirements, (language, terminal location and use, user/data classification) the total of the techniques listed opposite the three design features in the table are applicable for security protection.

Table 5. E-2 provides an example of applicable techniques for a system with no on-line language capability, local terminal with common access rights, and a single level of user/data clearance level.

TABLE 5.E-1. AUTOMATED TECHNIQUES RELATED TO DESCRIPTIONS OF SYSTEM DESIGN CHARACTERISTICS

System Design Characteristic	Techniques Applicable: Hardware Technique (H)	Techniques Applicable: Software Technique (S)
Class 1. User Language Capability		
• No interactive language	H32	S1, S3, S4, S12, S17, S24, S36, S40 and S41
• Fixed Transaction		S1, S2, S3, S4, S12, S17, S20, S24, S36 and S40
• Free-Form Query		S1, S2, S3, S4, S12, S17, S23, S24, S25, S36 and S38
• Assembly/POL	H31	S1, S2, S3, S4, S5, S6, S12, S16, S17, S18, S23, S24, S25, S35, S36 and S39
Class 2. Terminal Location and Usage		
• Local Terminals, Common Access Rights	H4, H17, H31 and H32	S3, S4, S24, S37, and S41
• Local Terminals, Different Access Rights	H4, H6, H17, H31 and H32	S3, S4, S9, S24, S25, S29, S30, S37 and S41
• Remote Terminals, Common Access Rights	H4, H6, H17, H29, H30 and H31	S3, S4, S8, S10, S11, S24, S31, S32, S37 and S41
• Remote Terminals, Different Access Rights	H4, H6, H17, H29, H30 and H31	S3, S4, S8, S9, S10, S11, S24, S25, S28, S29, S30, S31, S32, S37 and S41
Class 3. User and Data Clearance Level		
• Single Level		
• Mixed Levels	H6 and H20	S1, S9, S24 and S41
• Classified/Unclassified Levels	H6	S1, S9, S24, S29, S40 and S41

TABLE 5. E-2. EXAMPLE OF TECHNIQUES APPLICABLE TO A PARTICULAR CATEGORY OF SYSTEMS

System Design Characteristic	Techniques Applicable	
	Hardware	Software
o No Interactive User Language Capability	H32	S1, S3, S4, S12, S17, S24, S36, S40 and S41
o Local Terminals, Common Access Rights	H4, H17, H31, and H32	S3, S4, S24, S37, and S41
o Single Level of Uses and Data Clearance	-	-
o Sum of Automated Techniques Applicable to This Category	H4, H17, H31 and H32	S1, S3, S4, S12, S17, S24, S36, S37, S40 and S41

SECTION 6

DETERMINING TECHNIQUES APPLICABLE TO THE POST-1975 TACC

Section 6

DETERMINING TECHNIQUES APPLICABLE TO THE POST 1975 TACC

A. INTRODUCTION

The choice of applicable security techniques for the Post 75 TACC is based upon the selection of techniques that provides proper protection of classified data and processes without unwarranted overhead. The selection of applicable techniques depend upon the security requirements of the TACC, the determination of which techniques are applicable, and an estimate of the cost associated with the use of alternate techniques.

The techniques that are considered to be applicable to the TACC are a combination of those common to all automated systems and those necessary because of the particular design features of the TACC that influence security requirements. In this section the design features most probable for the Post 1975 TACC are identified, the security requirements identified and the applicable techniques are listed.

The selection of applicable techniques for the Post 1975 TACC proved to be particularly difficult because a methodology had to be developed to determine the class of security requirements that applied and a method developed to select applicable techniques based upon subjective evaluation of technique worth, qualitative cost estimates, and a subjective estimate of what is proper protection without unwarrented overhead cost. In addition, since the final design features of the Post 1975 TACC have not been selected, various combinations of security related design features had to be considered. Eight possible cases were selected and the security requirements and applicable security techniques for each case determined.

The section also includes comparisons between hardware, software, and procedural techniques and qualitative estimates of cost. They are included so that tradeoff between alternative techniques can be considered in the selection of applicable techniques.

B. TACC Security Requirements

The TACC has the security requirements that are common to all automated systems as discussed in Section 2; protection of data and processes from compromise through support access, failure access, electromagnetic radiation, and the planting of bugs. The unique security features of the TACC that influence security requirements is the user interface designed into the system to provide for its operational function. These are 1) language capability, 2) remote terminal use, and 3) user and data clearance levels. The eight possible combinations of these three features are shown in Table 6-1. The corresponding security requirements are listed by code number as defined in Section 2.

C. CATEGORIES OF TECHNIQUES APPLICABLE TO THE TACC

The complete set of techniques that are applicable to any system are those that are associated with the common categories of Support, Failure, Radiation, and Procedures, and those that are particular to the category of system characteristics to which the system is assigned. The TACC shares the first four subsets in common with every system, and these are shown in Table 6-2. Since the TACC may be assigned to any one of eight system categories, the techniques applicable to each of these eight are shown in Table 6-3.

TABLE 6-1. SECURITY REQUIREMENTS ASSOCIATED WITH THE POST-1975 TACC

Category of System Characteristic	Security Requirements
1. Fixed Transaction, Common Access, Mixed Clearances	R2-R5, R23-R34, R36-R43, R45-R53
2. Fixed Transaction, Common Access, Unclassified and Mixed Clearances	R2-R5, R23-R34, R36-R43, R45-R57
3. Fixed Transaction, Different Access, Mixed Clearances	R2-R5, R23-R43, R45-R53
4. Fixed Transaction, Different Access, Unclassified and Mixed Clearances	R2-R5, R23-R43, R45-R57
5. Free-form Query, Common Access, Mixed Clearances	R6-R12, R23-R43, R45-R53
6. Free-form Query, Common Access, Unclassified and Mixed Clearances	R6-R12, R23-R43, R45-R57
7. Free-form Query, Differenct Access, Mixed Clearances	R6-R12, R23-R30, R36-R43, R45-R53
8. Free-form Query, Different Access, Unclassified and Mixed Clearances	R6-R12, R23-R29, R36-R43, R45-R57

TABLE 6-2. SECURITY TECHNIQUES APPLICABLE TO EVERY SYSTEM

Category of System Characteristic	Security Techniques	
	Hardware Techniques (H)	Software Techniques (S)
• Support Access	H1, H2, H3, H6, H12, H16	S6, S7, S14, S15, S16, S17, S20, S35, S42
• Failure Control	H1, H2, H5, H7, H8, H16, H18, H24, H27, H33	S18, S23, S42
• Radiation	H29	S6, S31, S32, S39
• Procedures	P1, P2, P3, P4, P5, P11, P12, P13, P14, P19, P20	P6, P7, P8, P9, P10, P15, P16, P17, P18,

TABLE 6-3. SECURITY TECHNIQUES APPLICABLE TO CATEGORIES OF SYSTEM CHARACTERISTICS THAT DESCRIBE THE POST-1975 TACC

*Category of System Characteristic	Security Techniques	
	Hardware Technique (H)	Software Technique (S)
1.	H4, H6, H17, H20, H29, H30 and H31	S1, S2, S3, S4, S8, S9, S10, S11, S12, S17, S20, S24, S31, S32, S36, S37, S40 and S41
2.	H4, H6, H17, H29, H30, and H31	S1, S2, S3, S4, S8, S9, S10, S11, S12, S17, S20, S24, S29, S31, S32, S36, S37, S40 and S41
3.	H4, H6, H17, H20, H29 H30 and H31	S1, S2, S3, S4, S8, S9, S10, S11, S12, S17, S20, S24, S25, S28, S29, S30, S31, S32, S36, S37, S40 and S41
4.	H4, H6, H17, H29, H30 and H31	S1, S2, S3, S4, S8, S9, S10, S11, S12, S17, S20, S24, S25, S28, S29, S30, S31, S32, S36, S37, S40 and S41
5.	H4, H6, H17, H20, H29, H30 and H31	S1, S2, S3, S4, S8, S9, S10, S11, S12, S17, S23, S24, S25, S31, S32, S36, S37, S38 and S41
6.	H4, H6, H17, H29, H30 and H31	S1, S2, S3, S4, S8, S9, S10, S11, S12, S17, S23, S24, S25, S29, S30, S31, S36, S37, S38, S40 and S41
7.	H4, H6, H17, H20, H29, H30 and H31	S1, S2, S3, S4, S8, S9, S10, S11, S12, S17, S23, S24, S25, S28, S29, S30, S31, S32, S36, S37, S38 and S41
8.	H4, H6, H17, H29, H30 and H31	S1, S2, S3, S4, S8, S9, S10, S11, S12, S17, S23, S24, S25, S28, S29, S30, S31, S32, S36, S37, S38, S4C and S41

*The numbers refer to the category description in Table 6-1.

D. TRADEOFFS AMONG APPLICABLE TECHNIQUES

There are many more techniques applicable to the TACC than will be implemented. To aid the designer in making a tradeoff, a qualitative assessment of the cost of the techniques discussed is provided, as well as an evaluation of the relative merits of software, hardware, and procedural techniques.

1. Relative Costs of Techniques.

The 75 hardware and software techniques are summarized in this section in terms of their relative costs. These costs are divided into three areas. Cost for procedural techniques were not estimated.

- Response Time – the cost incurred by every message input to the system expressed as an effect on the length of time that a message response is delayed by the processing required for the technique in question.
- Throughput – the cost incurred by the system expressed as the decrease in the amount of processing the system is able to accomplish in a given time period caused by the additional processing required for the technique in question.
- Procurement – the costs associated with each technique expressed as the degree of expense involved in developing, maintaining, and servicing the technique in question.

The effect in each case is described as Low, Medium, or High, where Low is taken to mean less than a 5 percent increment in the cost of the system, Medium to mean between 5 and 10 percent, and High to mean more than 10 percent.

Table 6-4 (Software Techniques) and Table 6-5 (Hardware Techniques) present the techniques that might be used to protect against the listed security threats.

TABLE 6-4. SOFTWARE TECHNIQUE SUMMARY

Technique	Security Threat	Relative Cost		
		Response Time	Throughput	Procurement
S1. User classification	Identify user profiles	Low	Low	Low
S2. User access privileges	User accessing illegal data	Low	Low	Low
S3. Password	Not identifying users	Low	Low	Low-to-med.
S4. Password dialogue	Input of legal code by illegal person	Medium	Low	Low-to-med.
S5. Consecutive password list	Compromise identification of password	Medium	Medium	Low - med.
S6. Password transforms	Read other user passwords	Low	Low	Medium
S7. Digital checkpoint and hashing	Loss of data	Low-to-med.	Medium	Low
S8. Terminal answer-back	Piggyback terminals	Low	Medium	Low
S9. Terminal classification	Readout classified data at unclassified or lower classified terminal	Low	Low	Low
S10. Terminal character suppression	Inadvertent reading of synch keys	Medium	Low	Low
S11. Automatic alarm disconnect	Remote terminal interloper	Low	Med. -high	Medium
S12. Privileged instructions	Accessing illegal tables	Low	Low	Low

TABLE 6-4. SOFTWARE TECHNIQUE SUMMARY (Continued)

Technique	Security Threat	Relative Cost		
		Response Time	Throughput	Procurement
S13. Relocatable bootstrap	Read executive program by selected dumps	Low	Low	High
S14. Redundant coding	Modifying executive routines	Low-Med.	High	High
S15. Module dialogue	Accessing routines without privilege to do so	Low	High	High
S16. Program interpretation	Inputting illegal procedures or changes	Med-High	High	High
S17. Centralized I/O control	Bypassing table protection of data	Low-med	Medium	Low
S18. Error monitors	Failure in validity checks	Low	Medium	Medium
S19. Error interrupts	Invalid transmissions	Low	Low	Low
S20. Executive commands by access rights	Developing processes to access or change data	Low	Medium	Medium
S21. Boundary maps	Read/write outside of assigned area	Low	Low-Med.	Low
S22. Memory access keys	Inadvertant overwrites	Low	Low-Med.	Low
S23. Security monitor	Detect illegal entry attempts	Low-Med.	Medium	Med. High

TABLE 6-4. SOFTWARE TECHNIQUE SUMMARY (Continued)

Technique	Security Threat	Relative Cost		
		Response Time	Throughput	Procurement
S24. File classification code	Not distinguishing between several levels of classified data	Low	Low	Low
S25. File access lists	Unauthorized access	Low	Low	Low
S26. Deleted				
S27. File access lists by file	As above, but more granular	Low	Low	High
S28. File access profile	Remote user browsing in files	Low	Low	Low
S29. Data element classification	Mixed data classes	Med-High	Medium	High
S30. Data element profile	Multiple access privileges	Med-High	Medium	High
S31. File encryption-single key	Reading classified data without legal access	Low	Medium	Low-Med.
S32. File encryption-multiple key	Reading classified data without legal access	Low-Med.	High	Med.-High
S33. Data edition number	Mutual interference	Low	Low-Med.	Medium
S34. Block write-collision	Multiple concurrent users	Low	Med.-High	High
S35. Ring structures	Modify, read or execute illegal processes	Low-Med.	Medium	High

TABLE 6-4. SOFTWARE TECHNIQUE SUMMARY (Continued)

Technique	Security Threat	Relative Cost		
		Response Time	Throughput	Procurement
S36. Document log	Detect unauthorized input attempt	Low	Low-med	Low
S37. Limit number of erroneous attemps	Random input of combinations to obtain legal hit	Low	Med. -High	Med. -High
S38. Aggregate techniques for reports	Inference of higher level	Low-Med.	Low-Med.	Med. -High
S39. Overwrite and memory erase	Discovering classified residue	Medium	Med. -high	Low
S40. Classified programs	Access to generations routines	Low	Medium	Medium
S41. Classified headers/ trailers	Mixing classes of data	Low	Low	Low
S42. File log	Undetected accesses	Low	Medium	Low

TABLE 6-5. HARDWARE TECHNIQUE SUMMARY (CONTINUED)

Technique	Security Threat	Relative Cost		
		Response Time	Thruput	Procurement
H1. Processor Mode Privileged Inst. Set	Use of instructions which defeat software controls	Low	Low	Medium
H2. Core Memory Bounding	Addressing into blocks outside user program area	Low	Low	Medium
H3. Process Control Register	Attempts to execute unauthorized read/write/ executive instructions	Low	Low	Low
H4. I/O Control Registers and Mask Registers	Wrong channel I/O	Low	Low	Medium
H5. Parity Flag Bit Logic	Hardware error	Low	Low	Low
H6. Security Control Flag Bits	Classified data released to unclassified terminal	Low	Low	High – extra bits in each word for security
H7. Code Redundancy	Hardware error	Low	Low	Small to medium depending upon application
H8. Key Redundant Registers and Logic	Hardware error	Low	Low	Low to medium
H9. Parity Logic	Hardware error	Low	Low	Medium (extra bit for each word required)
H10. Key Word Register	Attempts to execute unauthorized read/write execute and to access outside user core area	Low	Low	Low
H11. Read Only Memory	Unauthorized alteration of programs or data	Low	Low	Medium

TABLE 6-5. HARDWARE TECHNIQUE SUMMARY (CONTINUED)

Technique	Security Threat	Relative Cost		
		Response Time	Thruput	Procurement
H12. Dedicated Memory	Unauthorized access or alteration to programs or data	Low	Low	Medium to high
H13. Memory Block Erase	Access to residue remaining in memory block	Medium	Medium	Low
H14. Associative Memory	Access to residue, unauthorized access to data programs, data destruction	Low	Low	High
H15. Memory Partitioning Ports	Unauthorized access to special programs or data	Low	Low	Low to medium
H16. I/O Bounds Control	Access to unauthorized main memory block	Low	Low	Low
H17. Unit Address Register	Misrouting of data	Low	Low	Low
H18. IOCP Parity Check	Hardware error	Low	Low	Low
H19. I/O Channel/No. Character Check	Misrouting of data	Low	Low	Medium
H20. I/O Security Level Register	Release of classified data on unauthorized channel	Low	Low	Low (high in memory requirement, see No. 6)
H21. Channel Number Check Logic	Misrouting of data	Low	Low	Low
H22. I/O Answer Back Check	Misrouting of data	Low to Medium	Low	Medium

TABLE 6-5. HARDWARE TECHNIQUE SUMMARY (CONTINUED)

Technique	Security Threat	Relative Cost		
		Response Time	Thruput	Procurement
H23. I/O Memory Erase	Access to residue remaining in memory block	Low	Medium	Low
H24. I/O Code Redundancy	Hardware error	Low	Low	Low to medium
H25. Read Only Lock	Unauthorized change to programs or data	Low	Low	Low
H26. Record Address Check	Misrouting of data	Low	Low	Low
H27. Data Parity Check	Hardware error	Low	Low	Low
H28. Check Sum Logic	Unauthorized change to data	Low	Low	Low
H29. Cryptographic Devices	Access to data by electro-magnetic radiation or wire tapping	Low	Low	High
H30. Hang Up and Dial	Misrouting of data	Low to Medium	Low	Low to medium
H31. Key Pattern Generator	Unauthorized access to programs or data	Low	Low	Medium to high
H32. Combination Lock or Lock and Key	Unauthorized access to equipment	Low	Low	Low
H33. Dual Hardware Access	Unauthorized access to equipment	Low	Low	Low
H34. Data Destruction Techniques	Area seizure	Low	Low	Medium to high

E. COMPARISON BETWEEN SOFTWARE, HARDWARE, AND PROCEDURAL TECHNIQUES

The comparison of hardware and software techniques identifies which security function can best be accomplished by each and whether a combination of both is necessary to provide adequate security of classified data. A comparison of manual versus automated systems procedures identifies the similarities between the two systems and the different approaches taken to perform the security function. Such a comparison provides a method to judge the relative value of automated techniques to achieve at least the same security level as manual systems. This paragraph considers first the comparison of hardware and software techniques which can be used for security purposes and then considers the similarities and differences between automated and manual system security procedures.

1. Hardware/Software Comparisons

The effective on-line control for the isolation of one user's programs and data from unintentional or intentional access or change by other user's programs can be achieved by hardware techniques. The use of a processor mode, privileged instruction set, and memory bounds provides the tools for effective isolation of programs and data. Without these hardware features, the software program would be required to examine and interpret each user program instruction in order to provide the same on-line control. The overhead involved in such software is estimated to run from 400-500 percent and is not considered practical. Even with these hardware features, the software executive system must be carefully designed so that circumvention of the hardware techniques is not possible. Other hardware techniques for which there are no potential practical software equivalents are; 1) the use of read/write/execute bit logic, 2) I/O control and mask registers with interrupt logic, and 3) hardware failure detection techniques. The on-line detection of unauthorized read/write/execute instructions again would require a software interpretor routine with resultant high overhead. The software alternative to I/O control hardware techniques is software polling routines which also have high overhead. The detection of hardware failures can be accomplished by on-line software diagnostic routines with resultant high overhead, but these are not as effective as parity check circuits. Fourth generation machines are proposing both hardware and software techniques to achieve higher system operating time.

The effective on-line control of files or data from unintentional or deliberate unauthorized access can best be achieved by software techniques. The use of user profile tables (which provide user code identification, data and program rights, and clearance level) provides an effective means to control access to programs and data. No hardware equivalent technique exists. Software programs can also provide security monitoring and security logging of all accesses or changes to data and programs. These have no equivalent technique in hardware.

Either software or hardware techniques can be used for communications channel coding of data, the clearing of residue data from main memory blocks, terminal user identification, detection of unauthorized change to data on direct access memory, and proper I/O routing. The selection depends upon cost effectiveness consideration in the system design of such factors as the number of channels, terminals, data base size, and CPU load.

Software and hardware techniques both are considered necessary for recovery from software failures and, if required, the effective isolation of support users' access to data and programs. Recovery from software failures, to be effective, requires

processor mode, privileged instruction set, and memory bounds hardware techniques plus software on-line diagnostics and a security monitor program. The isolation of support personnel from access to data and programs is the most difficult automated security technique to implement. The hardware techniques required consist of processor mode and privileged instruction set and could include the use of dedicated memory. Software techniques for isolation that have been suggested but not implemented include relocative bootstrap, redundant coding, module dialogue, and program interpretation.

A summary of the comparison of hardware/software techniques is given in Table 6-6. The table shows the security use, hardware and software technique applicable and remarks on major impacts.

2. Manual-Automated Procedural Comparisons

A comparison of procedures used in automated systems versus those that are used in manual systems provides a method to judge the relative value of automated techniques. This section will discuss those techniques that are common to both types of systems and those that are analogous.

In both automated and manual military systems procedural techniques are used to (1) secure areas where classified data is used, (2) to assure proper clearance of personnel, (3) to classify, access, disseminate, and control classified data, and (4) to protect classified data during transmission by cryptographic secure communications lines. The addition of automated techniques to increase the reliability of these procedures could be viewed as an attempt to increase the security of automated systems over that of manual systems.

Analogous techniques used in the two systems are (1) data storage procedures, (2) data access procedures, (3) data access accounting, (4) storage check procedures, and (5) inventory procedures. Data stored in a manual system is protected by three combination safe locks with 125,000 possible combination codes. In an automated system, data storage is based upon access codes with 4,000,000,000 possible combinations in a normal 32 bit code word. The probability of data access through other than code word knowledge is considerably less in an automated system than in the present manual system.

Data access procedures in a manual system are based upon access lists and personnel identification. In automated systems, access to data and files is based upon user/terminal profile tables and the requirement to submit the proper code word. Other techniques have been suggested such as finger print and voice code pattern generators.

Data access accountability in manual systems is performed by document sign-out. In automated systems, logs can be kept automatically of file access by user or terminal identification. Daily safe checks are used in manual systems to insure storage integrity. In automated systems, the access logs and security program reports can be reviewed daily or oftener if desired. Periodic inventory is used in manual systems to insure documents have not been lost or stolen. In automated systems, the files are reviewed periodically, check sum totals are used to insure data integrity, and all security logs reviewed.

The common and analogous techniques used in manual and automated systems are summarized in Table 6-7. The automated system, with the use of modest security techniques, can provide a greater level of security than is possible in a manual system.

TABLE 6-6. HARDWARE - SOFTWARE TECHNIQUES COMPARISON

Security Use	Automated Technique		Remarks
	Hardware	Software	
Limit input/output to executive software control.	Processor mode, privileged instruction set	Macro calls, edit programs to eliminate I/O instructions	Hardware technique provides cost-effective on-line check
Limit programs and data to an executive controlled memory block.	Memory bounds control	Boundary maps, edit programs to eliminate direct address use	Hardware technique provides cost-effective on-line check
Limit access to programs and data to predetermined read/write execute.	Read/write/execute bit logic. DAM read only lock	User rights profile table, edit programs to eliminate unauthorized calls	Hardware technique provides cost-effective on-line check
Provide for processing and proper routing of I/O requests.	I/O control and mask registers, interrupt logic	Software polling routine	Hardware techniques provide cost-effective on-line capability
Provide fail safe mode for hardware failures.	Parity check circuits, redundant key circuits, code redundancy	On-line software diagnostics	Hardware technique provides cost-effective on-line capability
Check security level of programs and data against access or processing request security level.	Security control flag bits	User rights profile table check	Software technique provides cost-effective on-line capability
Provide for insulation of user programs and data in main memory	Key word memory register, dedicated memory, memory partitioning ports, associative memory	Macro calls, edit programs to eliminate direct address use	Hardware techniques provide cost-effective on-line check

TABLE 6-6. HARDWARE - SOFTWARE TECHNIQUES COMPARISON (Continued)

Security Use	Automated Technique: Hardware	Automated Technique: Software	Remarks
Provide main memory residue data erase	Main memory block erase, IOCP block erase	Executive erases data before returning memory block for allocation	Hardware technique only necessary in CPU limited systems
Provide for proper routing of data in IOCP	Unit address register I/O channel/No. character check, channel number check logic, I/O answer back check.	Software answer back subroutine	Less processing overhead in hardware techniques. less procurement cost in software technique
Provide check of security level of data against security level of channel	I/O security level register	Channel profile table check	Software more cost-effective
Secure communications channels	Special purpose crypto device	Software crypto program	
Verify identification of terminal users	Hang up and dial, key pattern generator	Code word entry and check	Software less cost but not as reliable as hardware
Provide for detection of unauthorized change to data or programs	DAM check sum logic	Check sum software routine	More overhead for software routine
Provide for fail safe mode for software failure	Processor mode, privileged instruction set memory bounds	On-line diagnostics security monitor program	Both techniques are needed to provide effective control
Isolate support users from programs and data	Processor mode, memory bounds	Relocatable boostrap, redundant coding, module dialogue, program interpretation	Software techniques have high overhead

TABLE 6-6. HARDWARE - SOFTWARE TECHNIQUES COMPARISON (Continued)

Security Use	Automated Technique		Remarks
	Hardware	Software	
Limit access to files or data elements	Security control flag bits	File classification codes, file access rights, file access profile, data element classification code, data element profile	Software needed to provide effective control
Maintain security log		Document log, security program, file or data element log	No hardware technique available

TABLE 6.7. MANUAL/AUTOMATED SECURITY COMPARISON

COMMON

- Secured Area
- Personal Responsibility
- Security Regulations
- Standard Operating Procedures
- Communications Transforms

MANUAL		AUTOMATED
Combination Lock	versus	Passwords and Access Codes
Access Lists	versus	User/Terminal Profile Tables
Document Sign-Out	versus	File/Data Element Access Log
Daily Safe Check	versus	Access Log Review and Securiy Monitor Program
Periodic Inventory	versus	Record Counts, Check Totals, File Reviews

Appendix A

BIBLIOGRAPHY OF APPLICABLE DOCUMENTS

Appendix A

BIBLIOGRAPHY OF APPLICABLE DOCUMENTS

DOCUMENTS

1. AAS Implementation Manual, Security, Section V, IBM Internal Document 1133-2-3008, 9 April 1970.

2. AFM 207-1 TAC Sup I, Doctrine & Requirements for Security of Aerospace Systems, 2 April 1969.

3. AFM 207-21 TAC Sup I, System Security Standard, Command/Control Facilitites and Communications Elements, 7 October 1968.

4. AFR 205-6 TAC Sup I, Personnel Investigations, Security Clearances, and Access Authorizations, 16 January 1968.

5. AFR 205-34 TAC Sup I, Ultra Sensitive Position Program, 9 October 1968.

6. ALCOM System Description, Appendix I & II, File Structure, March 1970.

7. E. O. Andrews, An Annotated Bibliography: Protection of Secure Classified Information in Remote Access EDP, Lockheed Missile and Space Company, Palo Alto, California, Report No. LMSC-LS-67-14, April 1967.

8. A Report Bibliography, Computer Handling of Classified Material, DDC, January 1970.

9. J. D. Babcock, A Brief Description of Privacy Measures in the RUSH Time Sharing System, Proc. SJCC 1967.

10. P. Baran, On Distributed Communications, Security, Secrecy, and Tamper Free Considerations, RAND Corporation RM 3765-7R, August 1964.

11. H. W. Bingham, Security Techniques for EDP of Multi-Level Classified Information, RADC TR-65-415, December 1965.

12. W. S. Bates, Security of Computer-Based Information Systems, Datamation, May 1970.

13. A. R. Cenfetelli, Data Management Concepts for DOS/360 and TOS/360, IBM Systems Journal, Volume 6, No. 1, 1967.

14. CDC MARS III/MASTER Preliminary Reference Manual, Chapter 8, Privacy, C. D. C. Pub. No. 60279300, August 1969.

15. E. Cleadore, et al, Automatic Security Classification Study, RADC TR67-472, 1967.

16. E. V. Comber, Management of Confidential Information, Proc. FJCC 1969.

17. Control Data 3300/3500 Computer System Reference Manual, CDC, 1969.

18. D. J. Dantine, Communication Needs of the User for Management Information Systems, Proc. FJCC 1966.

19. Ruth M. Davis, Information Control in Command/Control Systems, New Perspectives in Organizational Research, P464-478, Wiley, 1964.

20. D. D. C. Bibliography on Information Sciences, Volumes I & II, D. D. C. No. AD-829-001 and AD-829-002, 1967.

21. D. D. C. Five Year Plan Study, Volume III, State-of-the-Art, S. D. C. Report No. TM-WD-268/003/00, August 1966.

22. Definition of Automated Internal Switching, Western Electric, AD-470-717.

23. R. M. Fano, Computers in Human Society - For Good or Ill, MIT Technology Review, March 1970.

24. GE 615/635 Information Systems Manual, GE, 1969.

25. C. Fanwick, Computer Safeguards: How Safe are They? SDC Magazine, July 1967.

26. C. Fanwick, Maintaining Privacy of Computerized Data, SDC SP2647, December 1966.

27. E. L. Feige and H. W. Watts, Protection of Privacy Through Microaggregation, Social System Research Institute, University of Wisconsin.

28. A. G. Fraser, User Control in a Multi-Access System, British Computer Journal, October 1967.

29. P. Frogg, et al, IBM System 360 Engineering, Proc. FJCC, 1964.

30. E. L. Glaser, A Brief Description of Privacy Measures in the Multics Operating System, Proc. FJCC, 1967.

31. R. M. Graham, Protection in an Information Processing Utility, Comm ACM 11, 5 May 1968.

32. A. Harrison, The Problem of Privacy in the Computer Age; an Annotated Bibliography, Volume 1, December 1967, RAND Corporation RM 5495 RR/RC; Volume 2, December 1969, RM 5495/1-RR/RC.

33. L. J. Hoffman, Computers & Privacy, A Survey, ACM Computing Surveys, Volume 1, No. 2, June 1968.

34. IBM System/360 System Summary, IBM Corporation, Form A22-C810-5, 1968.

35. Initial Systems Description, 7th Army Provisional Tactical Operating System, (TOS), DDC AD 381489, September 1966.

36. B. Litofski, Utility of Automatic Classification Systems for I. S. & R., University of Pennsylvania, Doctors Thesis, May 1969.

37. MIS Internal Specifications, Section 3. 0, Security Tables, IBM, June 1967.

38. E. Moore, Secure Communications, A Literature Survey, DDC Report No. TOR 269(9990)-5, November 1963.

39. NCIC Progress Report, FBI Law Envorcement Bulletin, September 1967.

40. NMCS NIPS/360 Manuals, General Description (Volume I) and File Structure (Volume II), 1968 Versions.

41. H. O'Neil, Security of ADP Information Systems, State-of-the-Art Applicable to the FADAP System, G. E. Tempo, April 1968.

42. B. Peters, Security Considerations in a Multi-Programmed Computer System, SJCC 1967.

43. H. E. Peterson and R. Turn, System Implications of Information Privacy, Proc. SJCC 1967.

44. Phase I Final Report, Marine Tactical Command and Control System Test Bed, Hughes Aircraft Company, 2 March 1970.

45. RADA System Design, Security: Some Techniques for Encryption of Messages for Transmission to Users, Martin Marietta, AD-392-425.

46. Report Bibliography, Computer Handling of Classified Material, DDC, October 1969.

47. Safeguarding Classified Information, AFR 205-1 up to Change No. 7, 24 June 1969.

48. Security and Privacy in a Common User Switching System, DCA Report No. 2424-64-19, July 1964.

49. R. O. Skatrud, A Consideration of the Application of Cryptographic Techniques to Data Processing, Proc. FJCC 1969.

50. System Security of the Defense Intelligence Agency's Analyst Support and Research System, Section III - Software Security, 28 August 1969.

51. TACM 100-2, Control of Compromising Emanations, January 1966.

52. TELTAP II, General Description, N. M. C. S. S. C. Report No. CSM-7A-67, March 1968.

53. The Computer and Invasion of Privacy, Hearings Before theSubcommittee of the Committee in Government Operations, House of Representatives, 89th Congress, 2nd Session, 28 July 1966.

54. UNIVAC AN/UYK-8(V) Military Computer, UNIVAC Division of Sperry Rand.

55. UNIVAC 1108 System Description, UNIVAC Division of Sperry Rand.

56. UNIVAC 494 Real-Time System, Sperry Rand Corporation, 1969.

57. D. A. Van Tassel, Advanced Cryptographic Techniques for Computers, Com of the ACM, December 1969.

58. D. A. Van Tassel, Cryptographic Techniques for Computers, Proc. SJCC 1969.

59. C. Vogt, Making Computerized Knowledge Safe for People, MIT Technology Review, March 1970.

60. W. H. Ware, Security and Privacy: Similarities and Differences, Proc. SJCC 1967.

61. W. H. Ware, Security and Privacy in Computer Systems, Proc. SJCC 1967.

62. C. Weissman, Programming Protection: What Do You Want to Pay? SDC Magazine, July 1967.

63. C. Weissman, Security Controls in the ADEPT-50 Time-Sharing System, Proc. FJCC 1969.

64. Xerox Data Systems Sigma 5/7 Manual, XDS, 1969.

65. Col. G. A. Zacharias, Guidelines for the Security of Automatic Data Processing (ADP) Systems, DOD Appendix to I. S. M., Draft, 14 January 1970, (Also, C. D. Ruyle, CODSIA Reply and Revisions), 5 March 1970.

Appendix B

SYSTEMS SURVEYED

Appendix B

SYSTEMS SURVEYED

SUMMARY

Twenty Government and fourteen commercial systems were surveyed to determine the security requirements and the security techniques employed by each system. A summary of the features of each system is presented in the following pages. The 34 systems are presented in alphabetical order by acronym, discussed in accordance with the following outline:

- Background
- System Security Requirements
- Hardware Configuration and Special Security Techniques
- Software Configuration and Special Security Techniques
- File Characteristics and Special Security Techniques

AAS

Background

The Advanced Administrative System (AAS) was developed by IBM's Data Processing Division as an internal software support product.

The system was designed to handle the administration of branch offices, including order entry and status, personnel, competitive data, prospect lists, etc.

System Security Requirements

The system must protect data and processes in an environment characterized by fixed-transaction interactive user language capability, remote and local user terminal locations, different user terminal access rights, and several levels of user clearance.

Hardware Configuration and Special Security Techniques

Six IBM 360/65s are configured as a multiprocessing system, three as transaction processors including Comm-switching, one as a Comm backup processor, and two as Data Management processors (one a backup of the others). 1×10^6 bytes of main memory are used. Ten IBM 360/30s act as line concentrators for 300 dedicated (but commercial) lines connected to 2,000 Model 2260 terminals. These terminals have ID blanking codes and hard-wired answerback codes. Forty-two 2314s provide 1.8×10^{10} bytes of secondary storage. One hundred fifty test sites have hardware voice ID key-pattern generators.

Software Configuration and Special Security Techniques

A greatly modified version of OS-360 HASP MVT with BTAM is used. New routines were added for terminal identification, logging, audit trail maintenance, error checking and switchover. Much of the unnecessary code in the original was pruned out. The system uses fixed transaction input. User profiles (currently 12,000 users) are maintained that identify ID, location, transactions authorized, and transactions qualified. User IDs are 4-cc alphamerics changed monthly or whenever a violation is suspected. They are assigned from a randomly generated list maintained by the Security Officer. The user logs in his ID plus payroll number, and logs out after each transaction. Two consecutive errors cause the terminal to be locked, upon which only the Security Officer's ID code can unlock it. Audit trail is maintained, indicating time, place, and personal ID for every input and update. System programmers cannot enter code from a terminal, cannot execute their own code, and cannot modify another's code. All support responsibility is fragmented. Separate logs are maintained of violations, erroneous inputs, and changes to access rights tables. Fifty modules (approx 4,000 bytes each) of code are required for Security routines. Eight man years of effort was required to develop the system. The Security function uses approximately 3 percent of system thruput.

File Characteristics and Special Security Techniques

System and Data files are basically structured using BDAM and ISAM. Access is controlled at the field level by assigning field IDs to transactions. User profiles are considered as separate data sets and require the full Model 2314 secondary storage. Access is limited to the Security Officer only.

ADEPT/TDMS

Background

ADEPT/TDMS was implemented by SDC of Santa Monica for ARPA and the U.S. Air Force Development Lab. The system was designed as a general purpose, on-line, resource-sharing system to handle sensitive information in classified Government and military command facilities.

System Security Requirements

The system must protect data and processes in an environment characterized by Assembly/POL interactive user language capability, remote and local user terminal locations, different user terminal access rights, and several levels of user clearance, including unclassified.

Hardware Configuration and Special Security Techniques

ADEPT/TDMS is currently implemented on an IBM 360/50 with 256,000 bytes of core. Fetch (or Read) protect has been added to the standard 360 hardware protection features to protect Executive areas in memory.

Software Configuration and Special Security Techniques

The ADEPT operating system is used. It has been extensively modified to operate in a secure environment. All users are identified by a unique 1-13 character alphameric ID. User profiles are maintained that describe the legal terminals available to the user, the files to which he has access, the quality of access (Read, Write, Read and Write, Lockout Override), and the list of legal user passwords (up to 64). The passwords are used only once in sequence and are then discarded. Terminal profiles and job profiles are also maintained. Both security level codes (0-4) and security category codes (0-16) can be assigned to any profile list. On-line debugging is permitted, but user files may only be read, and software checks are made on address references. All I/O is handled by a channel-compile routine and I/O supervisor. Only acceptable I/O references can be executed. Classified residue in core and on the drum is overwritten with 0's and user-specified digits. Disk storage is maintained "dirty." Special routines are available to the Security Officer for creating and modifying authority lists. These special routines required 5 percent of the development effort and 5,000 lines of code to implement. The effect on thruput is negligable except at File Open time.

File Characteristics and Special Security Techniques

Six categories of files are permitted: BSAM, ISAM, non-formatted, multi-volume fixed record, single-volume fixed record, and partitioned. Files can be public, private, or shared. If shared, a list of user IDs and access authority is maintained in the catalogue. Simultaneous access is permitted for Read only. ADEPT will attempt to automatically classify new files created from existing files by developing a composite ownership-level-category descriptor representing the most significant value of each in the contributing files. An audit-list of all file accesses, identifying user, terminal, time, and job is maintained.

AESOP

Background

The Advanced Experimental System for On-Line Processing (AESOP) was developed by the Systems Design Lab of MITRE Corp., Bedford, Mass. The system was designed to test on-line information handling concepts for management and command information systems.

System Security Requirements

The system must protect data and processes in an environment characterized by a free-form interactive user query language capability, local-only user terminal location, common user terminal access rights, and several levels (public and private) of user clearance.

Hardware Configuration and Special Security Techniques

AESOP was developed on an IBM STRETCH computer (7030) with 65,000 64-bit words of memory. 2.0×10^6 words of secondary storage are available on a 353 disk-unit. Four DD-13 Graphic Display consoles are connected on-line to the system. No special hardware features were implemented for security.

Software Configuration and Special Security Techniques

The AESOP system incorporated most of the security techniques that characterize 3rd generation operating systems. In addition, all displays are labled with their security classification, but no other software features were added for security purposes.

File Characteristics and Special Security Techniques

All files are tree-structured hierarchical blocks (pages). They can be considered to be private, but only as a processing convenience.

AFAL-RECON

Background

The AFAL-RECON system was developed for the Air Force Avionics Laboratory Reconnaissance Division, Wright Patterson Air Force Base, Ohio. The system was designed to provide automated indexing and retrieval of information on reconnaissance techniques. Users submit requests for data retrieval to support personnel by off-line requests. The system is operated in a batch job mode.

System Security Requirements

The system must protect data and processes in an environment characterized by local-only user terminal location, different user terminal access rights, and several levels of user clearance. There are no interactive user language capability requirements.

Hardware Configuration and Special Security Techniques

The IBM 360/40 hardware is used with job card input.

Software Configuration and Special Security Techniques

The standard batch IBM 0S/360 system is used. No modifications were installed for security purposes.

File Characteristics and Special Security Techniques

Batch oriented files under the IBM 0S/360 file system are used.

AFICCS

Background

The Air Force Integrated Command and Control System (AFICCS) was developed for the Headquarters, USAF, Washington, D.C. The system was designed for Air Force Headquarters level planning, summary reporting, and analysis.

System Security Requirements

The system must protect data and processes in an environment characterized by local-only user terminal location, different user terminal access rights, and several levels of user clearance (all except unclassified). There are no interactive user language capability requirements.

Hardware Configuration and Special Security Techniques

The system uses IBM 360/50 hardware with card input.

Software Configuration and Special Security Techniques

The system uses the IBM 0S/360 MVT software system as a card input off-line batch system. File access is restricted to those support personnel who submit a correct identity card.

File Characteristics and Special Security Techniques

A specially developed fixed-format file system is used. Access at the file level is limited to specified support personnel.

AFSS

This system is identical to the NSA system; refer to that heading for the discussion of security requirements and techniques.

ALCOM

Background

The ALCOM system was developed by the Applied Logic Company of Princeton, New Jersey. The system was designed as a general-purpose time-shared information processing utility system.

System Security Requirements

The system must protect data and processes in an environment characterized by Assembly/POL interactive user language capability, remote and local user terminal locations, different user terminal access rights and several levels of user clearance.

Hardware Configuration and Special Security Techniques

The system consists of a set of duplexed PDP-with high-speed drum storage that has been modified to permit multi-processing, segment addressing, and list-oriented instructions for queue manipulation. A special hardware technique for security purposes was implemented to permit class ϕ program users (system support personnel) to decode class ϕ routines.

Software Configuration and Special Security Techniques

Each user is identified by a 3-part 36-character alphanumeric branching code that names the user, within a project, within a company. A master file directory defining each valid user-file combination is maintained as a list of pointers to User File Directory entries. Access to files can be for Write, Overwrite (limited to user), List, or Lookup (available to project and company). The Master File Directory is in binary encrypted format and can only be accessed by class ϕ programs (which operate in privileged mode). A request for another user's file is only possible if the class ϕ program validates the link between the two User File Directory. All class ϕ code is maintained in encrypted form, with the key being generated in virtually a non-repeating manner.

File Characteristics and Special Security Techniques

All files are maintained in hierarchic fixed-length pages. They can be encrypted using a multi-key transformation algorithm (changing on every n-th record), but no user has exercised this option. Huffman encoding is used for compating transmitted data.

BTSS

Background

The Berkeley Time-Shared System (BTSS) was developed at the University of California, Berkeley, after the model of MIT's project MAC CTSS system. The system was designed to provide an information processing utility for the University's computing community, with facilities for simplified on-line interaction as well as conventional batch processing and programming.

System Security Requirements

The system must protect data and processes in an environment characterized by fixed-transaction interactive user language capability, local-only user terminal location, different user terminal access rights, and several levels (public and private) of user clearance.

Hardware Configuration and Special Security Techniques

The system uses a GE-635 computer with no special protection techniques.

Software Configuration and Special Security Techniques

The system uses MAC CTSS, modified to provide for an increased number of terminals.

File Characteristics and Special Security Techniques

The files can be encrypted using a single key for each file. In addition, both the user ID file and the file descriptors are organized in random fashion. An algorithmic generation of the user profile address from his input key and the data descriptor block address from the requested data elements permits rapid match of access rights.

DIA

Background

The DIA system was developed for the Defense Intelligence Agency, Washington, D. C. The system was designed for the analysis and summary reporting of intelligence data at the national level.

System Security Requirements

The system must protect data and processes in an environment characterized by a free-form interactive user query language capability, remote and local user terminal locations, different user terminal access rights, and several levels of user clearance, but not including unclassified.

Hardware Configuration and Special Security Techniques

Two GE 635 computer systems are used, one of which is for backup only. The direct-access memory consists of eight tape drivers and a Burroughs 270 disc with 15.3 million characters. Thirty-five terminals are connected, 10 CRT (Ratheon) and 25 TTY (37RO), through a Datanet 30 communication controller. Crypto-secure communications lines are used to connect the remote terminals. As prototype I/O channel/number of character check register was tested, but in over a year of use, no violations were detected and the device was dropped.

Software Configuration and Special Security Techniques

A modification of the Dartmouth College-developed time slicing Executive is used in the system. The GE GECOS III Executive system is presently being modified and will eventually replace the Dartmouth Executive. The system user language consists of BASIC and a specially developed query language, RAMAS (random access data base system).

An on-line security monitor is used that performs hardware/software checks of the system to ensure security-related controls are working. The porgram size of the monitor is 2,000 words. 110,000 hardware checks and 10,000 software checks are performed during each shift. Security-related failures have been identified only twice in the one year in use. An automatic terminal disconnect has been implemented if a code word is incorrect or if there is no access on RAMAS terminals in three minutes. BASIC users are allowed 30 minutes.

File Characteristics and Special Security Techniques

All files are 5,000 words per record or less and limited to a maximum of 1,000 records. Access is controlled by user identification and password to the file level. Read/Write control is maintained for each file by a table authority code. The access table consists of approximately 2,000 words.

DM-1

Background

The Data Manager-I system (DM-1) was developed by the Auerbach Corp., of Philadelphia, Pa. The system was designed as a general-purpose data management system capable of being implemented on most hardware systems and relatively independent of both Executive and Data characteristics.

System Security Requirements

The system must protect data and processes in an environment characterized by a free-form user query language capability, local-only user terminal location, different user terminal access rights, and several levels of user clearance.

Hardware Configuration and Special Security Techniques

The system is hardware independent, but generally requires about 65,000 lines of code resident in the configurations in which it is implemented.

Software Configuration and Special Security Techniques

DM-1 relies on the host operating system for protection against inadvertent compromise, although it does maintain its own task-data linkage lists. DM-1 will not permit direct user interface with Data files. A user profile keyed from the user password contains a list of the files in which Read and Write priveleges are assigned, and the level (0-7) to which they are permitted. A special language is available only to the Data Administrator to update item lists and user codes.

File Characteristics and Special Security Techniques

DM-1 files are either hierarchic packed bit-stream or random in organization. A Data Encoding Table gives a security class (0-7) for every data element. The combination of data elements required by a user for a job is checked when the data requests are linked to the data representations. The job is aborted if any element security level exceeds the users authorized clearance. A scheme of data element edition numbers has been proposed for permitting concurrent update of shared data files, but has not been implemented.

ESMIS

Background

The Employment Security Management Information System (ESMIS) was developed by the Auerbach Corporation under contract to the Federal Labor Bureau and the states of Michigan, Utah, and Florida. The system was designed to permit state employment agencies to match job requirements to applicant qualifications. The system must pro-

System Security Requirements

The system must protect data and processes in an environment characterized by fixed-transaction interactive user language capability, local-only user terminal location, different user terminal access rights, and several levels (public and private) of user clearance.

Hardware Configuration and Special Security Techniques

The system operates on an IBM 360/50 with twenty Model 2260 terminals and Model 2314 secondary storage. ESMIS requires a 65,000 byte partition and 512,000 bytes of library space on the 2314. No additional hardware features were implemented for security.

Software Configuration and Special Security Techniques

Standard OS-360 MVT BTAM is used except that ESMIS maintains its own file catalogue to increase thruput. ESMIS relies on terminal and user profiles for limiting access. Key to the user profile is a social security number input as ID. An ID/Write privilege level of four bits is associated with the user ID and with all data elements in the files.

File Characteristics and Special Security Techniques

All files in ESMIS are hierarchial packed bit-streams. Concurrent usage of data files is permitted, including concurrent updates. A block-lockout scheme is implemented to minimize file contention. Nearly 45 percent of the I/O coding was devoted to this routine.

FBI NCIC

Background

The FBI National Criminal Information Center (FBI NCIC) was developed by IBM and the FBI for the Department of Justice, Federal Bureau of Investigation, Washington, D. C. The system was designed to provide on-line criminal information to law enforcement agencies throughout the country. Want/warrants, stolen property, and gun registration information is available in the system files.

System Security Requirements

The system must protect data and processes in an environment characterized by fixed-transaction interactive user language capability, remote and local user terminal locations, common user terminal access rights, and only private clearance levels.

Hardware Configuration and Special Security Techniques

An IBM 360/65 computer system with on-line remote terminals is used. The terminals are connected by common commercial communications lines.

Software Configuration and Special Security Techniques

The IBM 0S/360 MVT software system is used. No special security techniques have been implemented.

File Characteristics and Special Security Techniques

The files are structured using BSAM and ISAM. No additional security techniques have been implemented to limit file access. Fixed format messages are processed and file access is based upon the requirement of the message input.

FTD

Background

The FTD system was developed for the Foreign Technology Division, Air Force Systems Command, Wright Patterson Air Force Base, Ohio. The system was designed for the indexing, surveying, translation, and summary reporting of foreign technical data. Users submit requests for services to support personnel by off-line requests. The system is operated in a batch job mode.

System Security Requirements

The system must protect data and processes in an environment characterized by local-only user terminal location, different user terminal access rights, and several levels of user clearance. There are no interactive user language capability requirements.

Hardware Configuration and Special Security Techniques

An IBM 360/50 computer system together with a specially designed disc system for automated language translation is used. Card input is used to define job sequences.

Software Configuration and Special Security Techniques

The standard batch IBM 0S/360 system is used. No modifications were installed for security purposes.

File Characteristics and Special Security Techniques

The system uses specially designed language translation files. The standard IBM 0S/360 file system is used for other data.

GIM

Background

The Generalized Information Management system (GIM) was developed by TRW Systems, Inc., Redondo Beach, California, as a commercial adaptation of the Cheyenne Program's ITDS. The system is designed as a general-purpose data management system capable of providing a shared data base to on-line users. It is intended to be generally independent of hardware and software characteristics.

System Security Requirements

The system must protect data and processes in an environment characterized by a free-form interactive user query language capability, local-only user terminal location, different user terminal access rights, and several levels of user clearance.

Hardware Configuration and Special Security Techniques

GIM is considered to be hardware independent, but generally requires 32,000 bytes of core, tape and mass storage secondary memory subsystems, and on-line terminal connection. It relies on the host hardware protection features.

Software Configuration and Special Security Techniques

GIM is considered to be independent of the host operating system, but will rely on it for general protection against inadvertent compromise. Internally, a comprehensive data audit is performed on all incoming data to validate field values; in addition, a log (historical trace) at every system input and change indicating time and point of origin, user ID, and current status of the data item is maintained for recovery. User profiles are maintained, keyed from the user ID Code. Permissible file access and the level to which access is permitted is described. The ID-level combination can be appended to system macro-commands and data correlations when these are link-edited so that a given user is restricted to only a defined subset of the processing capability.

File Characteristics and Special Security Techniques

GIM files are hierarchic variable-length pages. A security code can be assigned to individual data attributes, including particular values of these attributes.

GIS

Background

The Generalized Information System (GIS) was developed by the IBM Data Processing Division, White Plains, N. Y. The system was designed to provide for creating, storing, maintaining, retrieving, and reporting from a set of commonly accessed data files. No specific functional purpose is intended other than information storage and retrieval.

System Security Requirements

The system must protect data and processes in an environment characterized by local-only user terminal location, different user terminal access rights, a free-form interactive user query capability, and several levels of user clearance.

Hardware Configuration and Special Security Techniques

GIS operates on a minimum IBM 360/40 computer system with 131,000 bytes of core, decimal arithmetic, and direct access secondary storage. No additional features for security are provided.

Software Configuration and Special Security Techniques

GIS operates as a JOB under OS-360 and relies on the operating system for all terminal and process control protection. GIS will optionally make use of a category-security code (decimal number from 000-128) that can be assigned to any group or any specifically identified user. The code can be used to control both Read and Update privileges. Users are identified by an 8-digit alphanumeric code that is a key to a system maintained profile table. Access to files can be either partial or total; if partial, a generated list of allowable categories is used to control access.

File Characteristics and Special Security Requirements

GIS uses OS-360 file structure and access mechanisms (BSAM, ISAM, QISAM). Security category-codes can be assigned at the field-level through a system utility routine, *SEC. Both Read and Update codes can be assigned. A minimum security level can be assigned to the file itself that indicates the level of partial access permitted.

MARK-II

Background

The MARK-II system was developed and is marketed by the GE Time Sharing Services Division, Phoenix, Arizona. The system was designed to provide an information processing capability for a general class of remote batch-oriented users, although a limited conversational capability is provided.

System Security Requirements

The system must protect data and processes in an environment characterized by remote and local user terminal location, different user terminal access rights, and several levels (public and private) of user clearance. There are no interactive user language capability requirements.

Hardware Configuration and Special Security Techniques

MARK-II is currently implemented on systems with a duplexed pair of GE-635s, with GE-215 and DATANETS acting as line handling concentrators. All terminals in the system are equipped with a unique 20-digit answer-back code, and the system will only respond to valid inputs from coded terminals. Also, disconnect is automatic if no valid code has been entered within 30 seconds of log-in.

Software Configuration and Special Security Techniques

Terminal profiles are maintained that identify legal user passwords at each terminal. The GECOS III operating system is used to protect against unintentional compromises.

File Characteristics and Special Security Techniques

Ownership of files can be declared as either public or private. If private, the files can only be accessed by the individual that created them.

MARS

Background

The Multiple Access Retrieval System (MARS) was developed by CDC's Program Support Division, Lower 3000 Group, Minneapolis, Minnesota. The system was designed as the general-purpose operating and data management software package made available with the CDC 33/3500 system. It is a multiprogramming information storage and retrieval system, and no specific application function is intended.

System Security Requirements

The system must protect data and processes in an environment characterized by a free-form interactive user query language capability, remote and local user terminal locations, different user terminal access rights, and several levels of user clearance.

Hardware Configuration and Special Security Techniques

The system requires a minimum configuration of a CDC Model 3300 computer with 64,000 words of core and 1.2×10^7 characters of secondary storage. IBM 2250s, TTYs and CDC 217 displays are supported. The 3300 has paging registers that permit absolute boundary protection for multiprogramming, as well as privileged instructions, Read-protect, and Executive mode hardware.

Software Configuration and Special Security Techniques

This system takes full advantage of hardware paging on the 3300. In addition, the integrity of all task-to-task and task-to-data linkages is maintained. Library cataloging is available for program and data files. User profiles are maintained containing privilege words indicating need-to-know permission for Read or Write of data. The key to this file is an 8-character password. All legal account numbers and file privileges are described in the user profile. 3,000 words of core are required for validating user-entry. Terminal profiles and report profiles are also maintained, and any legal combination of user, terminal, and report can be specified. 7,000 words of code are implemented to define and maintain profile tables. The User Profile File is maintained as a MASTER file, however, and is accessible by any MASTER user.

File Characteristics and Special Security Techniques

MARS files are packed block data files with either list or paired index entries. Indexing is permitted on any field. Associated with every data element is both a Read and Write privilege word, permitting assignment of up to 64 levels of protection and 18 need-to-know flags. User ID is checked for correspondence to highest level and then to composite privilege word for each data request. Ownership of Data Profile file is restricted to security personnel. Affect on thruput is negligible if input requests are not recompiled, but are costly (a minimum of four device accesses/transaction) if requests are recompiled.

MIS

Background

The Management Information System (MIS) was developed by IBM from the FFS software implemented in NIPS. The system was designed as a general-purpose data management system to permit on-line usage of a common data base. No specific functional purpose is intended.

System Security Requirements

The system must protect data and processes in an environment characterized by a free-form interactive user query language capability, remote and local user terminal locations, different user terminal access rights, and several levels of user clearance.

Hardware Configuration and Special Security Techniques

The minimum system operates on an IBM System 360/40 with 256,000 bytes of core, a 2311 disk drive, and either a 2848 or 2702 for terminal control. No additional features for security are provided.

Software Configuration and Special Security Techniques

MIS operates as a JOB under OS-360 and utilizes the integrity features of that system. In addition, all users are identified to the system through a 3-6 digit alphameric code, which is system assigned and changed periodically. The code must precede every terminal entry; if nothing is transmitted in n-seconds, the terminal is dropped and must be reinitiated. The user-code is a key to the profile table which indicates the data and categories and clearance level the user is permitted. Access can be limited to any one of 10 levels of Read and Update privileges, and can also specify unique legal values permitted. Both the User Description Table and Access Description Table are maintained as separate data bases from the system file; a special entry is made in the UDT to limit access to these files to support personnel. 30,000 bytes of code were developed for creation and maintenance of the security tables (generally requiring 50-60,000 bytes plus one cylinder of 2314 for storage). The analysis of input messages required an additional 2,000-3,000 bytes.

File Characteristics and Special Security Techniques

The system uses standard OS-360 file structure and access mechanisms. However, security category codes (any 3 alphameric digits) and level codes (0-9) can be assigned to each data element. Certain values in a data field can be assigned limited access as well.

MULTICS

Background

The Multiplexed Information and Computing Service (MULTICS) was implemented by the Project MAC group at MIT, partially under contract to ARPA. The system was designed to provide information processing utility services to a large and varied group of MIT users.

System Security Requirements

The system must protect data and processes in an environment characterized by Assembly/POL interactive user language capability, remote and local user terminal locations, different user terminal access rights, and several levels of user clearance.

Hardware Configuration and Special Security Techniques

The system runs on a duplexed set of GE-645s. The I/O bus controller and the page address registers were modified to permit the introduction of segment addressing within a defined hierarchy of rings of protection. Segment references indicate ring brackets within which Read, Write, Executive, or Append (modify procedural segments) privileges can be allowed.

Software Configuration and Special Security Techniques

The system is a unique time-sharing executive and file handling utility, intended to permit multiple remote users concurrent access to the CPU. Protection against inadvertent compromise is achieved through the use of terminal address tables, cooperation with the segment registers, and fragmentation of module access privileges. The fragmentation is accomplished by defining a series of gradually more fundamental system states, known as rings. Access to any ring is controlled by maintenance of user profile tables, descriptor segments for processes, and Access Control Lists associated with each node in the hierarchy of rings. System control routines, support routines for the ACLs and the ACLs themselves are assigned to ring ϕ, and only specifically identified support personnel are permitted direct access to this level. Any attempt to access a segment at a different Ring Level creates a system fault and trap to the Supervisor.

File Characteristics and Special Security Techniques

MULTICS files are hierarchial list-structural files of fixed length segments. They can be declared as privately owned (only available to the creator), shared (a specific access list is declared), or public (at the ring-level indicated). In addition, variations of Read, Write, Execute, and Append privileges can be superimposed. The initial implementation of the hierarchy of data and processes defined by the Rings created sufficient confusion to lead most users to declare their information as public, effectively negating the protection mechanisms.

NASA COHERENT

Background

The NASA COHERENT system was proposed to the NASA Manned Spaceflight Technology Division, Cambridge, Mass, by Auerbach, Inc. The system is intended to be an information processing utility to process in flight NASA mission data.

System Security Requirements

If the system is implemented as proposed, it will be required to protect data and processes in an environment characterized by Assembly/POL interactive user language capability, remote and local user terminal locations, different user terminal access rights, and several levels of user clearance.

Hardware Configuration and Special Security Techniques

The hardware configuration is not yet determined. COHERENT will rely on the host hardware for protection mechanisms.

Software Configuration and Special Security Techniques

COHERENT will rely on the host operating system for protection from inadvertent compromise. In addition, the security features will include user profiles, data element access, security logs, and a security monitor that will monitor terminal activity.

File Characteristics and Special Security Techniques

The file characteristics are not yet determined, but an edition-numbering technique has been proposed to handle concurrent usage of data files.

NMCS/NIPS

Background

The National Military Command System (NMCS/NIPS) was developed by IBM for the Joint Chiefs of Staff Command Headquarters, Washington, D. C. The system was designed for national-level command and control of the Armed Forces. The system provides for the analysis, summary reporting, and planning for national Armed Forces utilization.

System Security Requirements

The system must protect data and processes in an environment characterized by local-only user terminal location, different user terminal access rights, and several levels of user clearance. There are no interactive user language capability requirements.

Hardware Configuration and Special Security Techniques

The system uses an IBM 360/50 computer system with batch job card input.

Software Configuration and Special Security Techniques

The system uses the IBM 0S/360 MVT software system as a card input off-line batch system. File access is restricted to those support personnel who submit a correct identity card.

File Characteristics and Special Security Techniques

A specially developed fixed-format file system is used. Access at the file level is limited to specified support personnel.

NSA

Background

The NSA system was developed for the Headquarters, Air Force Security Service, National Security Agency. The system was designed for the analysis and summary reporting of special data.

System Security Requirements

The system must protect data and processes in an environment characterized by Assembly/POL interactive user language capability, remote and local user terminal locations, common user access rights, and several levels of user clearance (but not including unclassified).

Hardware Configuration and Special Security Techniques

The system uses a UNIVAC 494 computer system with direct-access memory. On-line remote and local terminals are connected by crypto-secure communications lines.

Software Configuration and Special Security Techniques

The system uses a job entry orientated multiprocessing/multiprogramming software system. POL and Assembly language programs can be entered from terminals. The executive system was modified to prevent circumvention of memory partitioning techniques.

File Characteristics and Special Security Techniques

Access to files and programs is granted based upon terminal profile tables. Profile tables control the right to Read/Write in files and to Access and Execute programs.

NTDS

Background

The Navy Tactical Data System (NTDS) was developed for the U.S. Navy by the Hughes Aircraft Company, Ground Systems Group, Fullerton, California.

The system was designed for air space monitoring and aircraft direction for ship defense.

System Security Requirements

The system must protect data and processes in an environment characterized by fixed-transaction interactive user language capability, local-only user terminal location, common user terminal access rights, and several levels (Secret and Confidential) of user clearance.

Hardware Configuration and Special Security Techniques

The system uses a core-resident, AN/USQ-20 computer with local on-line displays refreshed from core. No direct access memory is included. No special security techniques were developed.

Software Configuration and Special Security Techniques

The system uses special purpose software to convert and display radar data, and to direct aircraft by digital data link. No special security techniques were developed.

File Characteristics and Special Security Techniques

The files are considered to be table-resident tables.

RAPID

Background

The Remote Access Personnel Identification System (RAPID) was developed for the U.S. Army PERINSCOM, Washington, D.C., by C.D.C. The system was designed for the on-line reporting, analysis, and summary of Army personnel actions.

System Security Requirements

The system must protect data in an environment characterized by a free-form interactive user query language capability, remote and local terminal locations, different terminal access rights, and several levels (Secret and Confidential) of user clearances.

Hardware Configuration and Special Security Techniques

The system uses a CDC 3300 computer system. On-line local and remote terminals are connected by crypto-secure communications lines.

Software Configuration and Special Security Techniques

The system uses the CDC MASTER Group System. Terminal profile is also maintained. The profile can be modified dynamically and alternate terminals can be specified. The terminal I/O codes are generated from a list of random numbers. If three illegal attempts are received from a terminal, the central site is notified. Approximately 300 words of core are required for security check.

File Characteristics and Special Security Techniques

File level and data element access is granted by user profile. The files are compressed but, are not transformed. The security for each data element can be assigned a level from 0-9. All accesses to classified data are logged, including the date, time, originator, and wording of the entry.

STRICOM

Background

The Strategic Intelligence Command system was developed by MITRE Corp. for the USAF Strike Command, Tampa, Florida. The system was designed to permit exercising Command and Control of Joint Tactical Forces. It provides for situation monitoring, planning, and control of these forces.

System Security Requirements

The system must protect data and processes in an environment characterized by a free-form interactive user language capability, local-only user terminal location, common user terminal access rights, and several levels of user clearance (not including unclassified).

Hardware Configuration and Special Security Techniques

The system uses an IBM 360/50 computer with on-line local terminals, model 2250. No additional security techniques were implemented.

Software Configuration and Special Security Techniques

The system uses a modified version of 0S/360 for protection against inadvertent compromises. No special security techniques were implemented.

File Characteristics and Special Security Techniques

The system provides fixedformat and variable length files. There are no restrictions on file access by terminal users, and no techniques for securing files (other than 0S/360 Catalogue Features) were implemented.

SAC-COM

Background

The SAC-COM system (465L) was developed for the Strategic Air Command, Command/ Control Systems, Omaha, Nebraska. The system was designed to permit exercising Command and Control of strategic forces. It provides for situation monitoring and command direction of these forces.

System Security Requirements

The system must protect data and processes in an environment characterized by fixed-transaction interactive user language capability, remote and local terminal locations, common user terminal access rights, and several levels of user clearance (not including unclassified).

Hardware Configuration and Special Security Techniques

The system uses an AN/APQ-32 dual computer system with tape and drum direct access memory. Remote on-line terminals and local on-line terminals, an on-line large screen display, and separate on-line communications processors are connected by crypto-secure communications lines.

Software Configuration and Special Security Techniques

The system uses a specially designed Executive system to provide batch queueing of incoming status messages and processing of local terminal requests. No additional security techniques are implemented.

File Characteristics and Special Security Techniques

File access is granted by terminal rights table. Once granted access to files is based upon the processing requirement. No additional security techniques implemented.

SEEK-DATA II

Background

The Seek-Data II System was developed by C.D.C. for the USAF Tactical Air Command, South East Asia.

The system was designed to provide first-level automation of tactical airlift planning and frag-order preparation for airlift and tactical air strike missions. It has been expanded to include first-level automation of situation reporting and near-real-time situation monitoring.

System Security Requirements

The system must protect data and processes in an environment characterized by fixed-transaction user language capability, remote and local user terminal locations, common user access rights, and several levels of user clearance.

Hardware Configuration and Special Security Techniques

The system uses IBM 360/50-1130 computers with tape and direct-access disc memory. Remote and local terminals (IBM 2260s and TTY) are connected on-line through crypto-secure communications lines.

Software Configuration and Special Security Techniques

The IBM 0S/360 MVT software system is used. Access to the system is determined by user password and terminal ID. Terminal profiles are maintained. Once access is authorized, the user can access any file authorized for that terminal.

File Characteristics and Special Security Techniques

Files are structured using 0S/360 file types. The degree of access is linked to the transaction requirements. Terminals are authorized specific transactions.

TACFIRE

Background

The Army Tactical Artillery Fire Control System (TACFIRE) was developed by Litton Corp. for the U.S. Army field command. The system was designed for the direction of Army tactical artillery fire.

System Security Requirements

The system must protect data and processes in an environment characterized by fixed-transaction user language capability, remote and local user terminal locations, common user access rights, and several levels (Secret and Confidential) of user clearance.

Hardware Configuration and Special Security Techniques

The system uses the Litton 3050 computer system, with remote and local on-line terminals. Forward observers are connected through unsecure on-line communications lines (input only). All other terminals are connected through crypto-secure communications lines.

Software Configuration and Special Security Techniques

The operating system provides the basic protection and integrity mechanisms common to most 3rd generation operating systems. File-level access is based upon terminal message input. A consecutive password list is checked before the system accepts an input message from the forward observer.

File Characteristics and Special Security Techniques

No security access control of files is exercised. The files are accessed based upon precompiled requirements of fixed-format messages input from terminals.

TIPI

Background

The Tactical Intelligence Processing Integration (TIPI) System Test Bed, TAC Headquarters, Virginia, and the National Military Command Center/NOMIS, Washington, D.C., both share the same software development goals.

The system purpose of NMCC/NOMIS are the same as NMCC/NIPS. The TIPI Test Bed was formed to establish automated concepts and procedures for the processing of tactical intelligence information. The system supports intelligence reporting, analysis, and summary reporting of tactical intelligence.

System Security Requirements

The systems must protect data and processes in an environment characterized by a free-form interactive user query language capability, local-only user terminal location, different user access rights, and several levels of user clearance (not including unclassified).

Hardware Configuration and Special Security Techniques

The systems use an IBM 360/50 computer with local on-line terminals.

Software Configuration and Special Security Techniques

The systems use the SDC-developed ADEPT/TDMS software package. The ADEPT Executive is a time-slicing Executive that allocates processing time in slices to each user in turn. The TDMS data management system provides on-line, free-form query capability. Access to the file level is controlled by User Profile Tables. The tables also are used to authorize Read or Write privileges in the files. Progressive user code words are used to identify system users. The security level of each user is checked against the security level of files and programs. Access is granted only when the security level of the user is equal to or higher than the file or program. Users are automatically disconnected from terminals if the terminal is not active. (See also ADEPT/TDMS).

File Characteristics and Special Security Techniques

The primary file structure used is a completely inverted file. Access rights are granted at the file level (see also ADEPT/TDMS).

TOS

Background

The Tactical Operations System (TOS) was developed by Bunker Ramo for the 7th Army, Europe. The system was designed as a test bed used to determine automated concepts and procedures for tactical command and control of a field army. The system is used for situation reporting, analysis, and summary reporting of the tactical situation.

System Security Requirements

The system must protect data and processes in an environment characterized by fixed-transaction user language capability, remote and local user terminal locations, common user access rights, and several levels of user clearance.

Hardware Configuration and Special Security Techniques

The system uses a CDC 3300/1700 computer with tape, drum, and disc storage. Remote on-line terminals are connected by crypto-secure communications lines.

Software Configuration and Special Security Techniques

A special-purpose Executive system is used that incorporates most of the integrity techniques common to 3rd generation operating systems. Terminal inputs are checked against a terminal profile table.

File Characteristics and Special Security Techniques

Access rights to files are determined by a terminal profile table. Read and Write authority at the file level is determined by the terminal profile table.

TSOS

Background

The Time-Shared Operating System (TSOS) was developed as a commercial service by RCA, Inc., Cherry Hill, New Jersey. The system was designed to provide an information processing utility for a general class of remote batch-oriented users.

System Security Requirements

The system must protect data and processes in an environment characterized by remote and local user terminal locations, different user access rights, and public and private levels of user clearance. There are no interactive user language capability requirements.

Hardware Configuration and Special Security Techniques

TSOS is implemented on a duplexed pair of 70-46s with 262,000 bytes of core and a 567 drum storage system. No special hardware features for security are available.

Software Configuration and Special Security Techniques

TSOS is a modification of the standard Spectra Series Operating System. User profiles are maintained by listing all files to which access is permitted either directly or through link editing. The user's account-number is algorithmicly modified to the profile address. A system log is maintained that indicates who initiated each change to the data base. The file catalogue creation and maintenance routines required 4,000 bytes of resident core and 25,000 bytes of overlay space on the drum.

File Characteristics and Special Security Techniques

All files are maintained in standard Spectra file formats. Since access status and permission levels are maintained in the Vol-ID for each file, considerable overhead is required to initially verify a user's request for data files (which can not be concurrently accessed)

USAF 407L

Background

The USAF 407L Control Reporting Center was developed by the Hughes Aircraft Co., Ground Systems Group, Fullerton, California, for the USAF Systems Command, Electronics System Division, Bedford, Mass.

The system was designed to perform air space monitoring and aircraft direction for air defense in a tactical environment.

System Security Requirements

The system must protect data and processes in an environment characterized by fixed-transaction interactive user language capability, local-only user terminal location, common user terminal access rights, and several levels (Secret and Confidential) of user clearance.

Hardware Configuration and Special Security Techniques

The system uses the Hughes-developed HM-4118 computer. It is a core resident system with local on-line display terminals. Displays are refreshed from core. There is no direct-access mass memory included. There are no parity checking or privileged instruction set in the system.

Software Configuration and Special Security Techniques

The system is provided with special purpose software for the conversion and display of radar data.

File Characteristics and Special Security Techniques

The files are considered to be core-resident tables.

USNLCC

Background

The U.S. Navy Landingship Control Center system (USNLCC) was developed by UNIVAC Corp. for the Pacific Fleet Programming Center. The system was designed as a test bed to establish automated procedures for the logistic control of material for ship loading and unloading. The system supports the analysis, planning, and summary reporting on material status.

System Security Requirements

The system must protect data and processes in an environment characterized by a free-form interactive user query language capability local-only user terminal locations, common user access rights, and several levels (Secret and Confidential) of user clearances.

Hardware Configuration and Special Security Techniques

The AN/USQ-20 computer (as in NTDS) is being used in the test bed. Discs and tapes have been added to support on-line files. On-line local terminals are connnected to the system. No additional security techniques have been added.

Software Configuration and Special Security Techniques

The system uses a special Executive system that, in addition to providing general integrity techniques, also controls access to files or data elements by a terminal profile table. The profile table identifies the right to Read or Write in specified files or data elements. A free-form query language is available to the terminal users to query and update authorized files and data elements.

File Characteristics and Special Security Techniques

Terminal access rights are granted at the data element level.

WWMCCS/EXEC 8

Background

This is a proposed extension of the UNIVAC 1108/EXEC VIII hardware/software system to meet the requirements of the World Wide Military Command and Control System (WWMCCS) specification. The system is proposed to be an upgrading of DoD fixed headquarters command and control systems. It would provide for on-line reporting, planning, analysis and summary reporting where installed.

System Security Requirements

The system must protect data and processes in an environment characterized by free-form interactive user query language capability, remote and local user terminal locations, different user access rights, and several levels of user clearance (not including unclassified).

Hardware Configuration and Special Security Techniques

The system will use the UNIVAC 1108 computer system with on-line and local terminals. Direct access memory will include tape, disc, and drum. Crypto-secure communications lines will connect remote terminals. No additional hardware techniques are proposed.

Software Configuration and Special Security Techniques

The Executive Eight software system on the 1108, which provides multiprogramming/multiprocessing capabilities, has been proposed. A user profile is maintained indicating which files can be accessed and the degree of access (7 levels and 16 categories of access). In addition, the user can be restricted to executing only predefined EXEC commands. Support personnel are prevented from processing their own internal interrupts. The system will only accept a single illegal entry from a terminal, and then will lock the terminal. Only the Security Officer can reinitiate it.

A user can request data of a higher classification than his own, but it will only be outputted at the central site, where procedural accommodations can be made. The entire set of modifications for security purposes require 500 words of resident core plus 7, 000 words of overlay space on the disk. EXEC file-handler will be extended by an additional 6, 000 instructions to handle the authorization and maintenance functions. The effect on thruput is estimated to be an increase of only 1 to 2 percent.

File Characteristics and Special Security Techniques

User passwords transformed internally are used to determine Read/Write access rights at the file level. Passwords also are used to determine rights to use programs. Storage overwrite of main memory core segments and disk segments is provided before the segment is reallocated to prevent residue readout.

APPENDIX C

POST-75 COMPUTER HARDWARE PREDICTED MEAN TIME BETWEEN FAILURES (MTBF)

Appendix C

POST-75 COMPUTER HARDWARE PREDICTED MEAN TIME BETWEEN FAILURES (MTBF)

The probability of the unintentional release of data depends to some extent upon the probability that key circuits within the computer hardware will fail. A measure of this probability is the predicted MTBF of military computer hardware that will be available in the post-75 time period. The probability that an unintentional release of data due to hardware failure will occur is less than the hardware MTBF because parity check circuits will detect some circuit failures and others will not result in data release. The following failure rates are predicted upon circuit MTBF's:

a. Monolithic Integrated Circuits

Single Simple Gate:	0.000012 failures/1000 hours
Flip Flop Average:	0.000007 failures/1000 hours
Analog IC Average:	0.00035 failures/1000 hours
Hybrid Current Drivers:	0.0002 failures/1000 hours

b. Connector Failure Rate: 0.00005 failures/1000 hours

c. Core Stack (16K): 0.0484 failures/1000 hours.

d. Registers Average 300 Gate Equivalent

The overall MTBF of a computer system with one CPU, 64K core, and four I/O channels is predicted to be 21 days as shown in Figure C-1. Figures C-2 through C-6 show the predicted MTBF for the major computer subsystems. The core memory failures and power circuit failures are six times the failure of other units. Table C-1 shows the most critical circuits within each major subsystem of the computer, the associated MTBF's and techniques that can be used to provide fail safe procedures. The use of power interrupt circuits can be used for recovery from power failures which occur at a predicted rate of one each four months. Parity check circuits on memory data registers can be used to detect core memory failures predicted to occur at the rate of one each four months.

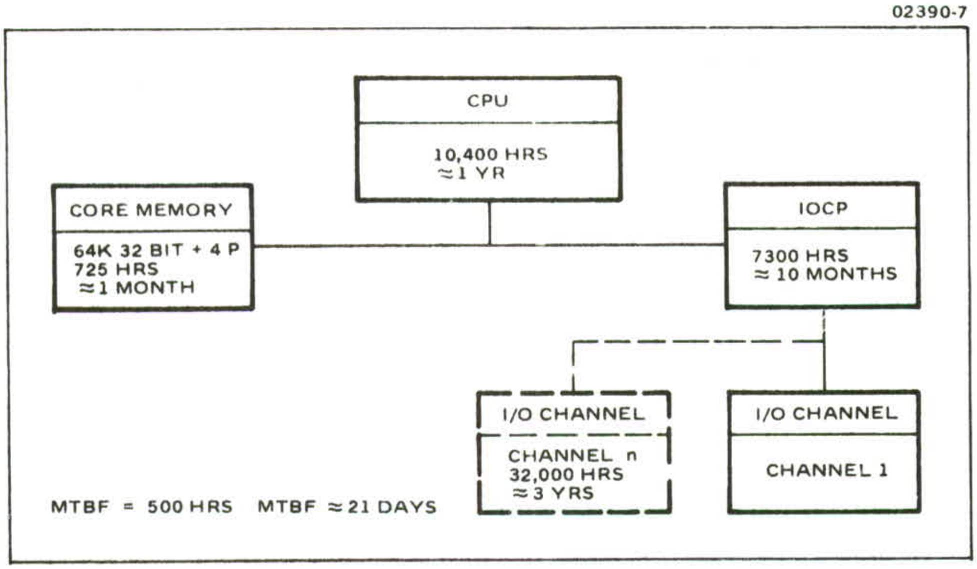

Figure C-1. Computer System Predicted Hardware MTBF

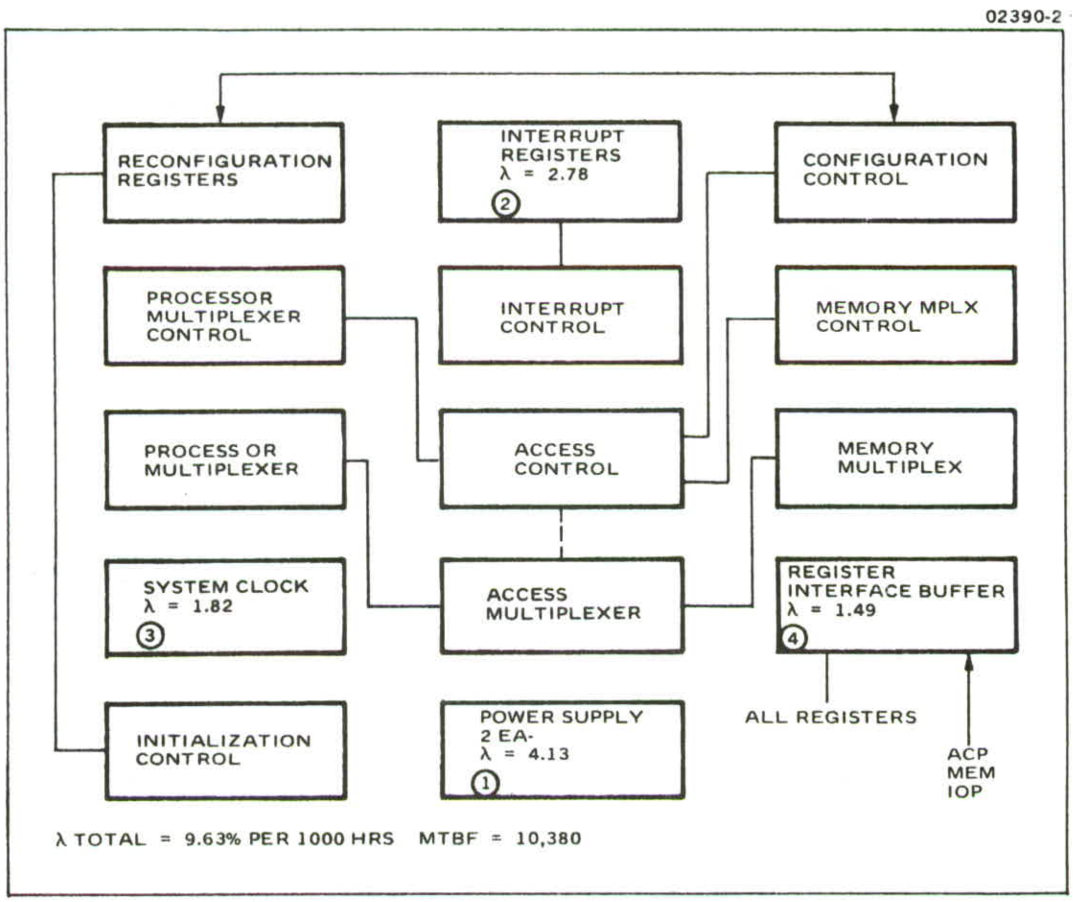

Figure C-2. Central Control Unit

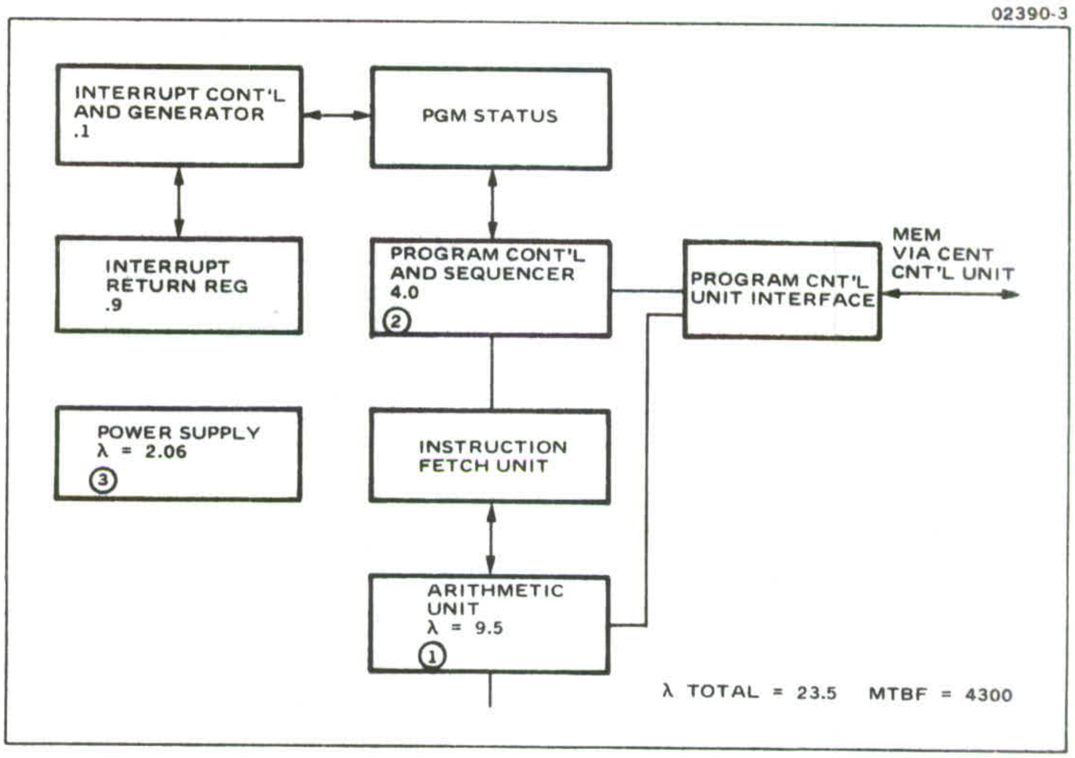

Figure C-3. Arithmetic Control Processor

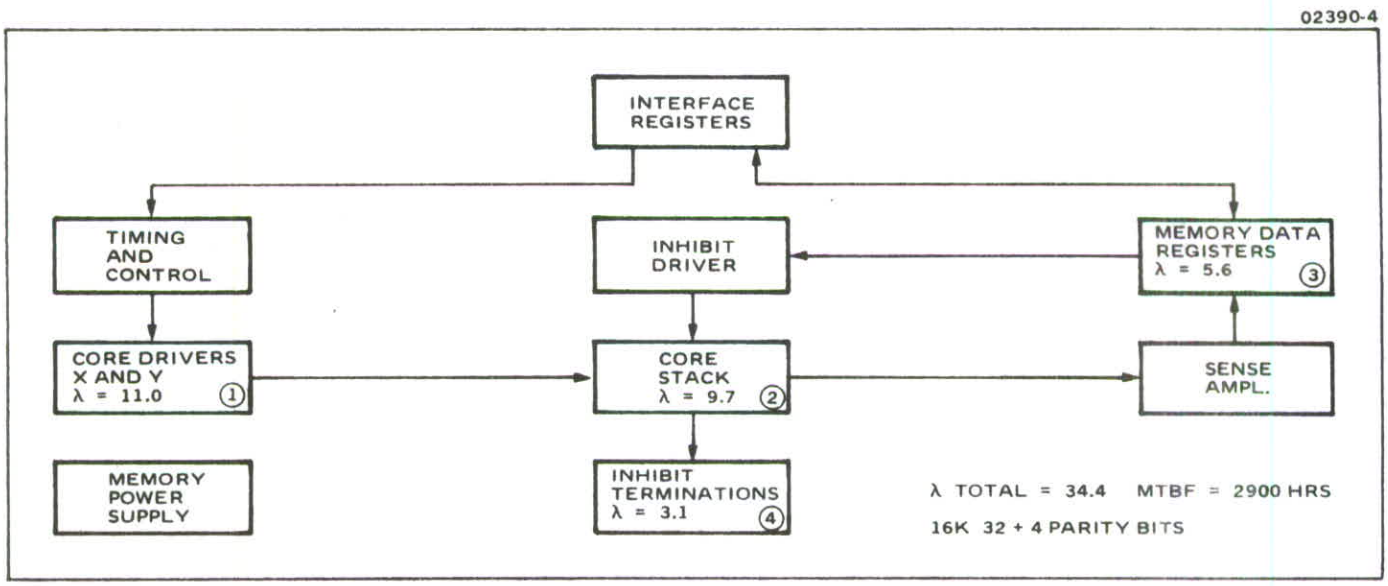

Figure C-4. Memory Unit

02390-5

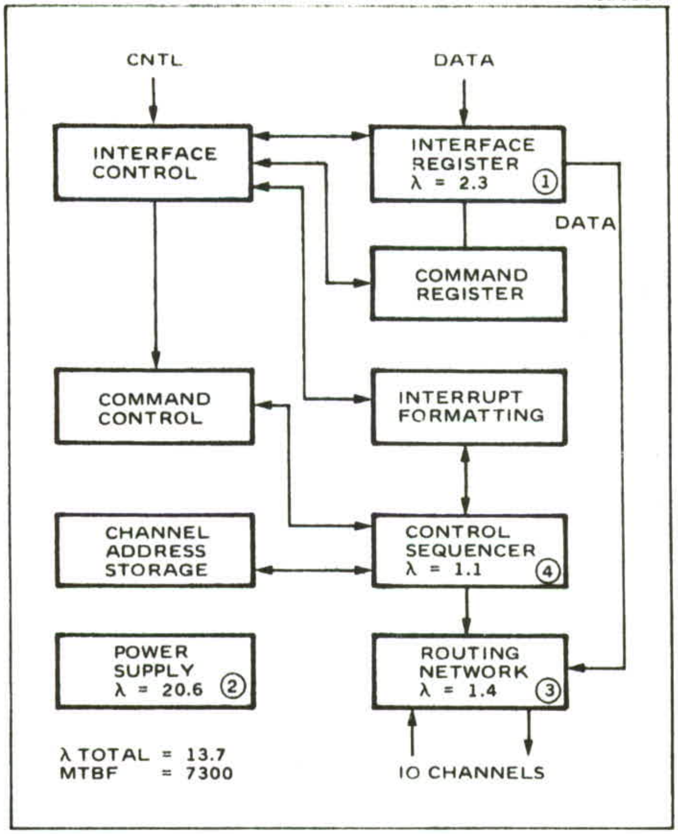

Figure C-5. IO Processor Common Circuitry

02390-6

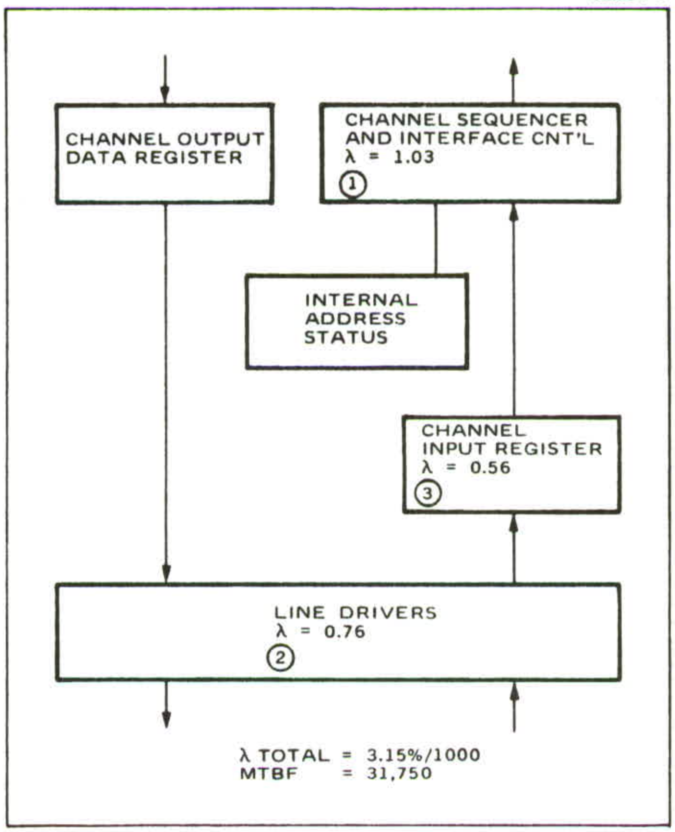

Figure C-6. Channel Electronics

Table C-1. Critical Circuits

Unit	Module	MTBF	Technique
Core Memory (64K)		3 Months	Parity Checks on Data Registers
	X and Y Drivers	4 Months	
	Core Stack	4 Months	
	Memory Data Registers	6 Months	
	Inhibit Terminations	6 Months	
CPU		5 Months	Separate Control Word Sequencing Parity Check Between Memory, CPU and I/O
	Arithmetic Control Unit	1 Year	
	Program Control Sequencer	2 Years	
	Interrupt Control and Register	4 Years	
	Data Interface Registers	4 Years	
Power Supplies		4 Months	Power Interrupt Circuitry
I/O Control		7 Months	Parity Checks (Security Level Check and Answer Back)
Data Interface Registers		4 Years	
Routing Network		8 Years	
Channel Interface(4 Channels)		11 Months	Parity Check
Channel Sequence and Interface Control		3 Years	
Line Drivers		4 Years	
Channel Input Registers		6 Years	

GLOSSARY OF TERMS USED

ACCESS CONTROL

The direct control of access to data in an automated system.

ACCESS LIST

The list of users or classes of users specifically granted access to data or processes.

ALPHANUMERIC

Character set including letters, numerals, and special characters.

ANSWERBACK

The acknowledgement of receipt of a transmission command, usually in the form of a hardware control word.

ASSEMBLY LANGUAGE

The set of alphabetic and numeric symbols that represent the machine language instructions and data in a computer. Alphabetic and numeric symbols are used in the same format as the machine binary code.

BROWSING

The attempt by an authorized user or by a penetrator who has been accepted as an authorized user to obtain, via an on-line terminal, data other than that authorized to the particular user.

BUGS

Recording devices used to collect and record electrical or acoustic data. They are the primary instruments of deliberate passive penetrations.

CATEGORIES OF SYSTEM SECURITY REQUIREMENTS

The node in a three-dimensional matrix of system design characteristics (User Language Capability, Terminal Location and Usage, and User Clearance Level) that implicity describes a set of security requirements; this set of security requirements then applies to any system having the associated design characteristics.

CENTRAL PROCESSING UNIT (CPU)

The subsystem of the computer that interprets program instructions and controls input/output of data.

CLEARANCE LEVEL

The maximum level of classified data to which an individual is granted access.

CLEARANCE PROCEDURE

The procedure wherein an individual's reliability is determined and access up to a given level of classified material is granted.

CODE WORD

A sequence of characters assigned to any system entity in order that individual entities can be positively identified to the system.

CONTROL MODE

Mode in which the processor can execute the full set of operation codes.

CRYPTOGRAPHIC TECHNIQUES

Data coding techniques commonly used to code data during transmission on a communications line. Hardware devices and software programs can be used to implement the techniques.

DATA BASE

The store of information records including programs being maintained for users.

DATA CLASSIFICATION

The level of security classification assigned to data.

DATA ELEMENT

The basic unit of data in a data base; it includes one unit of information.

DATA TRANSFORMS

Algorithms used to encode the normal representation of data codes in order to discourage access to the information content until decoding is accomplished.

DELIBERATE ACTIVE PENETRATIONS

Penetrations to obtain data that include deliberate intent and require activity against the system security techniques to accomplish.

DELIBERATE PASSIVE PENETRATIONS

Penetrations to obtain data that include deliberate intent and do not require activity against the system security techniques. They are, by their nature, undetectable.

DIAGNOSTIC PROGRAMS

Programs that detect and, in some cases provide for after-the-fact determination of, the causes of hardware and software failures in a system.

EXECUTIVE MODE

The same as control mode. Mode in which the processor can execute the full set of operation codes.

EXECUTIVE PROGRAMS

Programs that control the execution of user programs by assigning resources to them and performing security and/or data control operations as they are required.

FAILURE ACCESS

The unauthorized usually inadvertent access to data caused by hardware or software failure in the computer.

FAILURE CONTROL

The control methods used to detect and provide fail-safe (or fail-soft) recovery from hardware and software failures.

FAIL SAFE

The termination of programs or processing operations automatically when hardware or software failures are detected.

FAIL SOFT

The selective termination of affected non-essential processing after the cause of hardware or software failure has been diagnosed.

FIXED TRANSACTION

A preconceived fixed format input/output message that is assembled and transmitted when requested by a given code from a terminal. It may have varying values for the data elements, but cannot vary the data elements themselves.

FILE

A logical grouping of related records.

FLAG BIT

A bit contained in memory words and/or control words and used for control purposes.

FORMATTED FILE SYSTEM

An information storage and retrieval system using a file design having fixed, periodic and variable parts arranged in a hierarchic sequence described in a file format table.

FREE FORM QUERY

A data or program request that is not restricted to any predetermined data content and format, and which can therefore result in unpredictable access, assembly and transmission of data.

FUNCTIONAL SOFTWARE

The user programs that are designed to accomplish a particular mission requirement rather than the control of processing and data retrieval.

HALF-LIFE OF DATA

The time period from point of initiation during which data values describe the most current status of a system.

INPUT/OUTPUT CONTROL PROCESSOR (IOCP)

The subsystem of the computer serving as switching control between main memory, peripheral units, and terminals.

INTERRUPT

A signal that indicates the initiation or termination of certain system operations. It usually causes automatic switching to the executive or control mode.

KEY PATTERN GENERATOR

A transducer that generates a code word based upon an input such as voice, fingerprint, ID card, etc.

LIFE CYCLE COST

Cost associated with the procurement, training, operation and maintenance of an item over its useful life.

LOCAL TERMINAL

A terminal directly connected by cables without modems to the computer system; it is usually located in the same secure area as the computer system.

MACRO

A higher level computer language instruction that represents a commonly used logical process usually requiring several machine-level instructions to accomplish.

MAIN MEMORY

The storage media in which programs and data are stored and can be directly addressed by the CPU registers.

MASK REGISTER

A register that provides the CPU the capability to mask interrupts and control their processing.

MASQUERADING

The attempt by a penetrator to act as an authorized user of the system.

MEMORY BOUNDS

The establishment of a defined set of units of allocation in memory for the use of one user, and the limiting of all access to data and programs by that user to only these areas.

MODE

Processor condition as determined by state of a set of flip-flops.

MULTIPROCESSING

Executing one or more programs simultaneously on more than one central processing unit.

MULTIPROGRAMMING

Executing more than one program time-interleaved on the same central processing unit.

MULTIPLE ACCESS RIGHTS

The use accorded to the same terminal for more than one class of user with different data authorization or "need-to-know".

NEED-TO-KNOW

An expression of the group of data needed by an individual for the performance of his duties.

NON-FUNCTIONAL SOFTWARE

The computer programs in the system that provide for the control of processing and access to data.

NON-REVERSIBLE TRANSFORM

An algorithm for encoding data; it contains a substitution or transposition formula that produces stochastic results, or that has no mathematical or logical inverse.

OFF-LINE

No electrical terminal connection to the computer system.

ON-LINE

Direct electrical terminal connection to the computer system.

OVERHEAD

The amount of additional hardware capability or space, or processing time, or any other variable contributing to life-cycle cost needed to introduce a particular technique into a steady-state system.

PARITY

The addition of a bit to a memory word to indicate the value of the binary sum of the word bits. If the sum is odd, the parity bit is usually set to zero; if the sum is even, it usually is set to one (although this is reversed in many systems). Parity allows the determination of an odd (or even if the latter is true) number of bit errors in a memory word. Parity can also be used for parts of the memory word, or for longitudinal sequences of memory words.

PENETRATION

The act of obtaining data from a system without authorization.

PENETRATOR

The individual performing penetration of the system (also referred to as an "interloper" or "illegal agent").

PERIPHERAL UNIT

Devices other than terminals directly or indirectly, thru a multi-device controller, connected to the computer system through the IOCP.

POLLING

The scanning of devices in time sequence in order to determine their status.

PRIVACY

Security requirements for data and processes that stem from legal rights of ownership rather than from any defined set of security regulations; used synonomously in this report with 'Security'.

PRIVILEGED INSTRUCTION

An instruction which can only be executed by a CPU in control (or "executive") mode.

PRIVILEGED MODE

The CPU control mode that has the capability to execute all instructions.

PROCEDURE ORIENTED LANGUAGE (POL)

A computer programming language that uses macro instructions or application-oriented commands (e.g., FORTRAN-scientific, COBOL-business) requiring a compiler program to translate into executable machine code.

PROCESS CONTROL

The control of the execution of programs and use of physical devices in an automated system.

PROCUREMENT COST

The cost of initially procuring the hardware and/or software programs.

PROGRAM

The sequence of machine instructions that perform a logical process.

REMOTE TERMINAL

A terminal connected through a communications line with modems at either end to the computer center; it is normally on-line and normally located in a separate non-proximate area from the computer center.

RESIDUE

The data remaining in a portion of memory after the processing intended has been completed.

RESPONSE TIME

The time period involved from the receipt of the first character of a request for data until the first character of the requested data is made available.

SECURE AREA

An area where the right to enter has been limited to those authorized and is controlled by physical barriers and/or guard patrol.

SECURITY MONITOR

A software program that automatically monitors the security activity involved in an automated system.

SECURITY REGULATIONS

Formal directions for the establishment of security procedures to safeguard classified material.

SECURITY REQUIREMENT

A statement of the need to protect data or processes from illegal access; the individual security requirement is determined by the design characteristic being considered.

SECURITY RESPONSIBILITY

The formal doctrine that each individual is directly responsible for the proper protection of classified material that he has been granted access to for the performance of his duties.

SINGLE ACCESS RIGHT

Restricting the use of the same terminal to only one class of user with a single data authorization or "need-to-know".

SPECIAL DATA

Special classes of data that have limited dissemination to special (usually uniquely identified) users.

STRIKE OVER

The printing or display of a pattern character in every space of an area into which either the system or the user intends to enter sensitive information, the intention is to make the subsequent entry illegible to casual eavesdropping.

SUPPORT ACCESS

The access to data, programs, and equipment granted to system programmers and field service personnel in order to maintain an automated system.

TERMINAL

An input/output device that operates on-line. It could connect to the computer system either through communications lines or directly through the IOCP.

TERMINAL PROFILE

The list of access and process rights granted to a particular terminal.

THRUPUT

The number of tasks or jobs or the amount of processing, or any other measurable unit of information manipulation that can be accomplished in a given period of time.

14 KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Access Control						
Privacy						
Protection						
Security						
Security Requirements						